

# **Elastostatic Interaction Analysis of Frames Resting on Homogeneous Elastic Half-space**

**By**

**Reza Izadnegahdar**

**Submitted in fulfillment of the requirements  
for the Degree of Master of Engineering Science**

**University of Tasmania**

**February 2001**



UNIVERSITY OF TASMANIA

*Civil and Mechanical*  
School of Engineering

**Elastostatic  
Interaction Analysis  
of Frames Resting on  
Homogeneous Elastic Half-space**

**Master of Engineering Science Thesis**

Reza Izadnegahdar BSc(C/E)  
February 2001

## **ORIGINALITY**

This thesis contains no material which has been accepted for a degree or diploma by the University of Tasmania or any other institution, except by way of background information and duly acknowledged in the thesis, and to the best of the candidate's knowledge and belief no material previously published or written by another person except where due knowledge is made in the text of the thesis.

Signed:

A handwritten signature in black ink, consisting of a large, stylized 'R' followed by a series of loops and a final upward stroke.

(Reza Izadnegahdar)

## **AUTHORITY OF ACCESS**

This thesis may be made available for loan and limited copying in accordance with the Copyright Act 1968.

# ABSTRACT

The structural response of a building to applied loads depends on the behaviour of the soil supporting the structure. Structures may undergo various support deflections dependant on their support conditions. Such deflections are associated with soil deformation: the short-term deflections are known as the elastic component and, the long-term deflections are known as the permanent plastic component. Differential support displacements cause structural interaction between supports.

To analyse such interaction, there are different approaches; it can be analytical or numerical, static or dynamic and deterministic or probabilistic, with the rigour in the analysis being commensurate to the degree of displacement likely to be experienced. Mathematical rigour, however, may or may not be justified if inadequate knowledge of parameters exists. In the context of the natural variability of constituent parameters, a closer examination of the soil parameters associated with this interaction, particularly for plane frame structures, is warranted.

The behaviour of soil media at a structure's supports during structural analysis has been the focus of much investigation and research over the last century. Efforts have been made to describe such interaction and many computer packages have been developed to incorporate different soil models with the structural studies. Unfortunately the models are usually too complex or require too much input data for easy use by professional engineers.

This study focuses on the interaction context through a linear elastostatic and deterministic analysis of plane frame structures with different soil idealisations. The study concentrates on the behaviour of a space frame structure that is constructed by a number of typical plane frames in a parallel series subjected to individual loads. A homogeneous elastic half space is utilised for the soil support.



A displacement-type analytical-numerical technique of elastic solution is used to evaluate support interaction. In this approach, the behaviour of soil is modelled by means of homogenous, elastic half-space, whilst in the analysis of the structure, a Direct Stiffness Method is applied.

The Boussinesq and Cerruti force-displacement solutions are used for isotropic behaviour, whilst the Gerrard-Wardle and Gerrard-Harrison models are used for cross-anisotropic behaviour. The flexibility matrix of the elastic half-space, related to the interaction forces is developed. The author has prepared an integrated software program to perform the analysis on a desktop PC.

The interaction results of the support were analysed for sensitivity to the soil parameters. A number of typical soil parameters were considered for Winkler, isotropic and cross-anisotropic soil models in the structural analysis. Analysis outputs were shown in graphs and tables and were used to investigate for sensitivity of interaction to different soil parameters. Finally, the conclusion of the research work is drawn and the recommendation for further research study is suggested.

## NOTATIONS

$f_{ij}^{ab}$	Flexibility coefficient with superscripts $a$ and $b$ denoting the points of the displacement considered and of the load applied, and where the subscripts $i$ and $j$ designate the directions associated with the displacement and the load, respectively.
$A_1, \dots, A_9$	Coefficients' positions in an off-diagonal block in a soil flexibility matrix associated with two distinct points
$B_1, \dots, B_9$	Coefficients' positions in a diagonal block in a soil flexibility matrix associated with two coincident points
$x, y, z$	Local Cartesian coordinates
$u, v, w$	Displacements in $x, y$ and $z$ directions in subscripts presenting the location of the point considered
$X, Y, Z$	Global Cartesian coordinates
$r, \theta, z$	Cylindrical coordinates (radial, tangential and vertical directions)
$a, b, c, d, f$	Components of elasticity tensor for a cross-anisotropic material
$g_1, \dots, g_4, h_1, \dots, h_{11}, i_1, \dots, i_{11},$ $j_1, \dots, j_{11}, s_1, \dots, s_9, t_1, \dots, t_{10},$ $I_{200}, {}_c I_{200}, {}_s I_{200}, I_{202}, I_{222}, I_{220},$ ${}_c I_{220}, {}_s I_{220}, I_{420}, {}_c I_{420}, {}_s I_{420}, I_{422},$ $L_{000}, {}_s L_{000}, {}_c L_{000}, L_{002}, L_{020},$ ${}_c L_{020}, {}_s L_{020}, L_{022}, L_{120},$ $A_{220}, {}_s M_{220}, M_{222}, M_{420}, {}_c M_{420},$ ${}_s M_{420}, P_{020}, S_{020}, S_{022}, S_{120}, S_{122}$	Constants and integration coefficients in Chapters 2 and 3
$E_v, E_h,$ $F_v,$ $\nu_h, \nu_{hv}, \nu_{vh}$	Young's moduli, shear modulus, and Poisson's ratios in a cross-anisotropic soil material with symmetry about a vertical axis in its elastic properties
$E, G, \nu$	Young's and shear moduli and Poisson's ratio for an isotropic material
$E_s, G_s, \nu_s$	Young's and shear moduli and Poisson's ratio for an isotropic soil
$P_z, Q_x, M_y$	Concentrated vertical and horizontal forces and a horizontal moment
$p$	Vertical distributed load
$p_{\max}$	Maximum value for a linearly distributed vertical load
$q_x$	Horizontal distributed load

$T$	Tension field in a fictitious elastic membrane in two-parameter soil model
$D_p, D_r$	Flexural rigidity of a foundation plate of general shape, and a circle
$E_p, G_p, \nu_p$	Young's and shear moduli and Poisson's ratio for a plate
$r_0$	Radius of circular loading area
$k_s$	Modulus of sub-grade reaction in Winkler model
$k, K$	Transform parameters ( $K = k r_0$ Chapter I)
$f_a(r), f_b(r), f_c(r)$	Functions of $r$ corresponding to load stress distributions
$\alpha, \beta, \omega, \gamma, \phi, \rho, \psi, \Psi$	Derived elastic quantities reflecting nature of anisotropy
$H_0(k), H_1(k)$	Hankel transforms of order zero and one
$A_{\zeta\tau\lambda}(\Psi), I_{\zeta\tau\lambda}(\Psi), {}_c I_{\zeta\tau\lambda}, {}_s I_{\zeta\tau\lambda}, L_{\zeta\tau\lambda}(\psi), {}_c L_{\zeta\tau\lambda}, {}_s L_{\zeta\tau\lambda}, M_{\zeta\tau\lambda}(\Psi), {}_s M_{\zeta\tau\lambda}, P_{\zeta\tau\lambda}(\psi), S_{\zeta\tau\lambda}(\psi), {}_c S_{\zeta\tau\lambda}, {}_s S_{\zeta\tau\lambda}$	Integrals involved in the general solutions for displacements, strains and stresses
$l, A, I$	Length, cross-sectional area, the moment of inertia about the z-axis for structure element
$[K_{st}]$	Structure stiffness matrix
$[K_s]$	Soil stiffness matrix
$[K_{sys}]$	System stiffness matrix
$k_{ab}$	Element stiffness coefficient
$[k_{ab}]_{(e)}$	Stiffness matrix (3×3) for element (e) associated with forces at node $a$ due to displacements of node $b$
$[k]_e$	Local Element stiffness matrix
$[k]$	Global element stiffness matrix
$\{\bar{F}\}_e$	Element nodal force vector
$\{\Delta\}_e$	Local element nodal displacements vector
$\{\Delta\}_e$	Global element nodal displacements vector

$\{\Delta\}$	Nodal displacements vector
$[T]_e$	Element transformation matrix
$[T]_e^T$	Transpose of element transformation matrix.
$\{P\}$	Nodal external forces vector
$\{P\}^0$	Total fixed end forces vector
$\{R_s\}$	Reaction forces vector
$\{R_{rs}\}$	Nodal elastic restraint forces vector
$\{F\}_e$	Total local element end forces vector
$\{F^0\}_e$	Vector of local fixed end forces due to the loads applied to an element
$\{F^0\}_e$	Vector of global fixed end forces due to the loads applied to an element
$\{P\}_{eff}$	Vector of global effective forces in a system $\{P\} - \{P^0\}$

# CONTENTS

ORIGINALITY .....	i
AUTHORITY OF ACCESS .....	i
ABSTRACT .....	ii
ACKNOWLEDGMENTS.....	iv
NOTATIONS .....	v
CONTENTS .....	viii

## CHAPTER 1

INTRODUCTION .....	1
1.1 THESIS OUTLINE .....	1
1.2 STATE OF THE ART: SOIL-STRUCTURE INTERACTION .....	3
1.3 SIGNIFICANCE OF SOIL CONSIDERATION .....	4
1.4 LITERATURE REVIEW .....	6
1.4.1 Rigid Model.....	9
1.4.2 Winkler Model and Its Applications.....	10
1.4.3 Homogeneous Isotropic Elastic Half-space Model.....	13
1.4.4 Homogeneous Cross-anisotropic Elastic Half-space Model.....	16
1.4.5 Non-homogeneous Isotropic Elastic Half-space Model .....	17
1.5 REQUIREMENTS FOR SOIL-STRUCTURE ANALYSIS .....	19
1.6 SCOPE OF THIS THESIS .....	19
1.7 POSSIBLE BENEFITS OF THE RESEARCH .....	20

## CHAPTER 2

SOIL IDEALISATION.....	22
2.1 INTRODUCTION.....	22
2.2 SINGLE-PARAMETER SOIL MODEL .....	22
2.3 TWO-PARAMETER SOIL MODELS .....	25
2.3.1 Refinements to Winkler Model.....	25
2.3.2 Homogeneous Isotropic Elastic Half-space .....	26
2.3.2.1 Further Solution Associated with Isotropic Elastic Half-space.....	29
2.4 CROSS-ANISOTROPIC ELASTIC HALF-SPACE MODELS.....	32
2.4.1 Points of Load Application and Displacement Distinct.....	35
2.4.1.1 Displacements due to a Vertical Point Load.....	36
2.4.1.2 Displacements due to a Horizontal Point Load.....	37
2.4.1.3 Displacements due to a Moment Load about Horizontal Axis	
Z .....	37
2.4.2 Points of Load Application and Displacement Coincident.....	38
2.4.2.1 Displacements due to a Vertical Uniform Load.....	39
2.4.2.2 Displacements due to a Horizontal Uniform Load .....	40
2.4.2.3 Displacements due to a Vertical Linear Load .....	40
2.5 NON-HOMOGENEOUS ISOTROPIC ELASTIC HALF-SPACE	
MODEL.....	41

2.6 METHODS OF OBTAINING SOIL PARAMETERS .....	44
2.7 CONCLUDING REMARKS TO CHAPTER.....	49

### CHAPTER 3

<b>METHOD OF SOLUTION .....</b>	<b>50</b>
3.1 INTRODUCTION.....	50
3.2 STRUCTURE-SOIL SYSTEM.....	50
3.3 DIRECT STIFFNESS MATRIX METHOD.....	53
3.4 SUPERSTRUCTURE CONTRIBUTION IN ANALYSIS .....	54
3.4.1 Assumptions in Superstructure analysis.....	55
3.5 SOIL CONTRIBUTION IN ANALYSIS.....	55
3.5.1 Assumptions in Soil Consideration.....	56
3.5.2 Development of Soil Stiffness Matrix .....	56
3.5.3 Flexibility Coefficients Associated with Two Distinct Points on a Homogeneous Isotropic Elastic Half-space .....	58
3.5.3.1 Flexibility Coefficients Associated with Horizontal Force $Q_b$ .....	59
3.5.3.2 Flexibility Coefficients Associated with Vertical Force $P_b$ .....	61
3.5.3.3 Flexibility Coefficients Associated with Moment $M_b$ about Horizontal Axis $z$ .....	63
3.5.4 Flexibility Coefficients Associated with Two Coincident Points on a Homogeneous Isotropic Elastic Half-space .....	64
3.5.4.1 Flexibility Coefficients Associated with Horizontal Force $Q_b$ .....	65
3.5.4.2 Flexibility Coefficients Associated with Vertical Force $P_b$ .....	66
3.5.4.3 Flexibility Coefficients Associated with Moment $M_b$ about Horizontal Axis $z$ .....	67
3.5.5 Flexibility Coefficients Associated with Two Distinct Points on a Homogeneous Cross-anisotropic Elastic Half-space.....	70
3.5.5.1 Flexibility Coefficients Associated with the Displacements due to Horizontal Load $Q_b$ .....	71
3.5.5.2 Flexibility Coefficients Associated with the Displacements due to Vertical Load $P_b$ .....	72
3.5.5.3 Flexibility Coefficients Associated with the Displacements due to Moment $M_b$ about Horizontal Axis $z$ .....	73
3.5.6 Flexibility Coefficients Associated with Two Coincident Points on a Homogeneous Cross-anisotropic Elastic Half-space.....	74
3.5.6.1 Flexibility Coefficients Associated with the Displacements due to Horizontal Force $Q_b$ .....	75
3.5.6.2 Flexibility Coefficients Associated with the Displacements due to Vertical Force $P_b$ .....	76
3.5.6.3 Flexibility Coefficients Associated the Displacements due to Moment $M_b$ about Horizontal Axis $z$ .....	78
3.6 CONCLUDING REMARKS TO CHAPTER.....	80

### CHAPTER 4

<b>SOFTWARE DEVELOPMENT .....</b>	<b>81</b>
4.1 INTRODUCTION.....	81
4.2 MODULE "DATAENTR" .....	82
4.3 MODULE "ANALYSIS" .....	84
4.4 OUTPUT OF ANALYSIS .....	87

4.5 CONCLUDING REMARKS TO CHAPTER.....	87
<b>CHAPTER 5</b>	
<b>EXAMPLES OF ANALYSIS APPLICATIONS .....</b>	<b>88</b>
5.1 INTRODUCTION .....	88
5.2 HINGE-SUPPORTED FRAME AND SOIL MODEL APPLIED .....	88
5.3 EXAMPLES STUDIED IN THIS PROJECT .....	90
5.3.1 Nomenclature Applicable to Examples, Charts, Graphs and Tables .....	92
5.3.2 Examples of Analysis of Frame Founded on Isotropic Soil.....	99
5.3.3 Set One Examples for Cross-anisotropic Soil.....	106
5.3.4 Set Two Examples, Simulated Cases of Cross-anisotropic Soil .....	113
5.4 STRUCTURE FOUNDED ON AN INFINITELY RIGID SOIL .....	117
5.5 STRUCTURE FOUNDED ON WINKLER SOIL.....	117
5.6 VERIFICATION OF RESULTS OF ANALYSIS .....	120
5.7 JUSTIFICATION OF THE TYPE OF FRAME AND LOADS USED IN THE ANALYSES .....	121
5.8 CONCLUDING REMARKS TO CHAPTER.....	128
<b>CHAPTER 6</b>	
<b>DISCUSSION OF RESULTS .....</b>	<b>129</b>
6.1 INTRODUCTION .....	129
6.2 OBSERVATIONS AND DISCUSSIONS ON RESULTS OF ANALYSING FRAME ON ISOTROPIC ELASTIC HALF- SPACE.....	129
6.2.1 Case One of an Isotropic Soil .....	130
6.2.2 Case Two of an Isotropic Soil .....	134
6.2.3 Comparison of Cases One and Two of the Isotropic Soil .....	135
6.3 OBSERVATIONS AND DISCUSSIONS ON RESULTS OF ANALYSING FRAME ON CROSS-ANISOTROPIC ELASTIC HALF-SPACE .....	136
6.4 FINDINGS AND DISCUSSION ON RESULTS OF ANALYSIS: WINKLER MODEL AS COMPARED WITH ISOTROPIC ELASTIC HALF-SPACE.....	145
6.5 SENSITIVITY ANALYSIS .....	147
6.6 CONCLUDING REMARKS TO CHAPTER.....	148
<b>CHAPTER 7</b>	
<b>SUMMARY AND CONCLUSIONS.....</b>	<b>150</b>
7.1 SUMMARY .....	150
7.2 CONCLUSIONS .....	151
7.3 FUTURE WORK .....	153
<b>REFERENCES .....</b>	<b>R-1</b>
<b>APPENDICES</b>	
<b>APPENDIX A1.....</b>	<b>1-1</b>
A1 VERTICAL DISPLACEMENTS UNDER A LOAD DISTRIBUTED OVER A CIRCLE AND A RECTANGLE .....	1-1

<b>APPENDIX A2.....</b>	<b>2-1</b>
A2.1 HOMOGENEOUS ELASTIC HALF-SPACE.....	2-1
A2.1.1 Isotropic Elastic Half-Space .....	2-1
A2.1.2 Cross-anisotropic Elastic Half-Space.....	2-2
A2.1.2.1 Dirac Delta Function.....	2-3
A2.1.2.2 Constants and Coefficients Associated with Numerical Integration Method .....	2-4
A2.2 METHODS OF OBTAINING SOIL PARAMETERS .....	2-10
A2.2.1 Determination of Elastic Parameters of an Isotropic Soil.....	2-11
A2.2.2 In situ Determination of CBR and Elastic Parameters .....	2-11
A2.2.3 Laboratory Determination of CBR and Soil Elastic Parameters .....	2-13
A2.2.4 Adoption of Presumptive CBR Values .....	2-16
 <b>APPENDIX A3</b>	
A3 DEVELOPMENT OF STRUCTURE STIFFNESS MATRICES & DIRECT STIFFNESS METHOD .....	3-1
 <b>APPENDIX A4</b>	
USER MANUAL FOR SASIAP ANALYSIS PACKAGE .....	4-1
A4.1 Introduction .....	4-1
A4.2 Data Menu .....	4-2
A4.3 Edit menu .....	4-5
A4.4 Load Menu .....	4-8
A4.5 Computer Program Listings .....	4-12
Program GENFORMS.....	4-12
Unit GLOBALS.....	4-14
Program DATAENTR.....	4-21
Program ANALYSIS.....	4-44
Unit BUILD.....	4-64
Unit SOILMODE .....	4-80
 <b>APPENDIX A5.....</b>	<b>5-1</b>
A5.1 Tables associated with Graphs 5.1 - 5.18.....	5-1
A5.2 Tables associated with Graphs 5.19 - 5.36.....	5-7
A5.3 Tables associated with Graphs 5.37 - 5.45.....	5-13



# LISTING OF FIGURES

Figure 1.1	Flow chart illustrating the thesis layout.....	1
Figure 1.2	Classification of soil models and methods of stress consideration and calculation .....	7
Figure 1.3	Illustration of sign convention for a beam and a plate foundation resting on Winkler model.....	11
Figure 1.4	Illustration of soil deformations: (a) under external load for to Winkler model, and (b) the observed deflections on site in the real situation.....	11
Figure 2.1	Illustration of Boussinesq's approach over a semi-infinite soil mass bearing a normal load .....	27
Figure 2.2	Illustration of Cerruti's case over a semi-infinite soil mass bearing a normal load.....	28
Figure 2.3	Cylindrical coordinates and an infinitesimal circular loaded area (radius $r_0$ ).....	35
Figure 2.4	Illustration of load cases considered by Gerrard and Wardle (1973).....	35
Figure 2.5	Stress resolution for the load cases considered in this study (Gerrard and Harrison 1970a) .....	38
Figure 2.6	Variation of shear modulus in an inhomogeneous soil idealised by Gibson (1967).....	42
Figure 3.1	Coordinate systems, superstructure, soil medium and the interaction.....	52
Figure 3.2	Illustration of soil and superstructure nodal reactions at the interface .....	55
Figure 3.3	Illustration of different sub-matrix groups in soil flexibility between two points.....	56
Figure 3.4	Orientation of direction vector for point $P(i, j, k)$ .....	57
Figure 3.5	Flexibility coefficients associated with the horizontal and vertical degrees of freedom at point $b$ due to horizontal force $Q_a$ at point $a$ .....	57
Figure 3.6	Illustration of flexibility coefficients in soil flexibility matrix $[F]_{ab}$ .....	58
Figure 3.7	Profile of the surface displacement components due to horizontal surface force $Q_b$ .....	60
Figure 3.8	Equivalent situations in flexibility coefficients due to symmetry in the soil flexibility matrix .....	60
Figure 3.9	Resolution of moment $M_a$ into a vertical couple acting on the surface of the medium.....	60
Figure 3.10	Profile of the surface horizontal and vertical displacements due to vertical force ( $P_b$ ) .....	61
Figure 3.11	Rotations $\phi_a$ and $\phi_c$ due to the vertical point load $P_b$ .....	62
Figure 3.12	Replacement of moment $M_b$ with a couple, and the vertical displacement .....	63
Figure 3.13	Total zero vertical displacement due to horizontal load $Q_b$ .....	65
Figure 3.14	Load distribution over a circle on the soil replacing moment $M_b$ .....	69
Figure 3.15	Load distribution over a circle on the soil replacing moment $M_b$ .....	78
Figure 4.1	Overview of SASIAP analysis package .....	82
Figure 4.2	Flow chart for module "DATAENTR" .....	83
Figure 4.3	Flow chart for module "ANALYSIS" .....	84
Figure 4.4	Flow chart for soil section in the soil-structure analysis .....	85
Figure 4.5	Flow chart for structure consideration in analysis procedure.....	86
Figure 4.6	Flow chart for the system analysis .....	86
Figure 5.1	Illustration of a single plane frame considered for two cases: (a) No interactive support, and (b) With interactive support.....	89
Figure 5.2	Illustration of a typical (main) plane frame and neighbouring parallel frames .....	89
Figure 5.3	Profile of the structure and the external applied loads.....	90, 106 & 113
Figure 5.4	Flow chart for soil data in the examples.....	90

Figure 5.5	Nomination of symbols to element end-node moments associated with plane frame resulting from structural analysis.....	93
Figure 5.6	Nomination of symbols to nodal displacements associated with analysis.....	94
Figure 6.1	Illustration of parameters hierarchy in system analysis.....	148
Figure A1.1	Parameters used for a distributed load over a circle.....	1-1
Figure A1.2	Surface vertical displacement due to a uniformly loaded rectangular area on elastic half space (Cheung and Zienkiewicz 1965).....	1-2
Figure A2.1	Illustration of the Mindlin's case in x-z plane within the medium.....	2-1
Figure A2.2	Stress definitions in triaxial test .....	2-15
Figure A3.1	Element free-body diagram in local coordinates.....	3-1
Figure A3.2	Stiffness matrix for a structure with two members and three nodes.....	3-3
Figure A3.3	Orientation of displacements and forces in both coordinate systems.....	3-3
Figure A3.4	Orientation of fixed-end forces in the local coordinates .....	3-5
Figure A4.1	Illustrates the introduction of DataEntr program.....	4-2
Figure A4.2	A blank text associated with a default title.....	4-3
Figure A4.3	The Data menu bar and its pull-down items.....	4-3
Figure A4.4	Accessing and opening an existing structure data file.....	4-3
Figure A4.5	Illustration of window <i>Save As</i> to save a file under a name other than the original .....	4-3
Figure A4.6	Illustrates the root and current directories for any new location of a directory .....	4-4
Figure A4.7	DOS shell access within the program.....	4-4
Figure A4.8	Illustrates the items on the Edit data pull-down menu .....	4-5
Figure A4.9	Window associated with the title of a structure.....	4-5
Figure A4.10	Selection of a data file associated with Geometry of Structure nodes.....	4-6
Figure A4.11	Geometry details of a structure node entered or edited .....	4-6
Figure A4.12	Window to access a file of material properties.....	4-6
Figure A4.13	Illustration of material properties data window.....	4-6
Figure A4.14	Allocating a cross section data file.....	4-7
Figure A4.15	Cross-section properties associated with structure members .....	4-7
Figure A4.16	Allocating a member data file .....	4-7
Figure A4.17	Data associated with members of a structure .....	4-7
Figure A4.18	Allocating a file for Restraint data .....	4-8
Figure A4.19	Information associated with node restraints .....	4-8
Figure A4.20	Load pull-down menu .....	4-8
Figure A4.21	Allocating Prescribed Load file.....	4-9
Figure A4.22	Details of prescribed displacement loads applicable to structure nodes.....	4-9
Figure A4.23	Allocating Node load file .....	4-9
Figure A4.24	Details of Node loads applicable to structure.....	4-9
Figure A4.25	Allocating file associated with member concentrated load .....	4-10
Figure A4.26	Details of member concentrated load .....	4-10
Figure A4.27	Allocating file associated with member uniformly distributed loads.....	4-10
Figure A4.28	Details of member uniformly distributed loads.....	4-10
Figure A4.29	Allocation of file associated with trapezoidal loads.....	4-11
Figure A4.30	Details of trapezoidal loads applicable to structure.....	4-11

# LISTING OF TABLES

Table 2.1	Equations defining stress-strain modulus $E_s$ by several test methods .....	45
Table 2.2	The most commonly used multiplier $\sqrt{OCR}$ (overconsolidation ratio) used to increase the normally consolidated value of stress-strain modulus $E_{s,nc}$ .....	46
Table 2.3	Values and some value ranges for Poisson's ratio $\mu$ for selected materials.....	46
Table 2.4	The most commonly used values for soils.....	47
Table 2.5	A range of $E_s$ values (possibly obtained) for the static stress-strain modulus $E_s$ for selected soils.....	47
Table 5.1	Table of soil elastic properties considered in this research.....	95
Table 5.1a	Isotropic elastic soil constants (Gerrard 1967) .....	96
Table 5.1b	Table of properties for cross-anisotropic soil Gerrard (1967) .....	97
Table 5.1c	Listing of elastic properties for Isotropic and Cross-anisotropic soil sub-cases 32-36 (Wardle 1977).....	98
Table 5.1d	Listing of elastic properties for Isotropic soil sub-cases 37, 39, 40 and its equivalent of Winkler spring.....	98
Table 5.2	Output data for element bending moment $M_{21}$ from structure analysis when soil modulus of elasticity (Young's modulus ) varies between 5 and 150 $MPa$ for three different interaction cases.....	99
Table 5.3	Element end moments .....	117
Table 5.4.	Nodal displacements in $x$ and $y$ directions are in meters, and rotation in $z$ direction is in radian.....	117
Table 5.5	Support reactions in $x$ and $y$ directions are in kN, and for $z$ direction is in kN.m.....	117
Table 5.6	Bending moment $M_{21}$ for different Winkler springs and corresponding Isotropic values .....	118
Table 5.7	Bending moment $M_{32}$ for different Winkler springs and corresponding Isotropic values .....	118
Table 5.8	Bending moment $M_{34}$ for different Winkler springs and corresponding Isotropic values .....	118
Table 5.9	Nodal displacement $\Delta_{x1}$ for different Winkler springs and corresponding Isotropic values .....	119
Table 5.10	Nodal displacement $\Delta_{y1}$ for different Winkler springs and corresponding Isotropic values .....	119
Table 5.11	Nodal rotation $\phi_{z1}$ for different Winkler springs and corresponding Isotropic values .....	119
Table 5.12	Nodal displacement $\Delta_{x3}$ for different Winkler springs and corresponding Isotropic values .....	119
Table 5.13	Nodal displacement $\Delta_{y3}$ for different Winkler springs and corresponding Isotropic values .....	119
Table 5.14	Nodal rotation $\phi_{z3}$ for different Winkler springs and corresponding Isotropic values .....	120
Table 5.15	Nodal reactions from interaction analysis (Isotropic $E_s = 50 MPa$ , $\nu = 0.25$ ) .....	122
Table 5.16	Members end moments (kN.m) from interaction analysis (Isotropic $E_s = 50 MPa$ , $\nu = 0.25$ ) .....	122

Table 5.17	Nodal displacements ( $m$ ) Angle of rotation (rad) from interaction analysis (Isotropic $E_s = 50$ MPa, $\nu = 0.25$ ).....	122
Table 5.18	Verification of displacements for sample: 7-Iso D.....	123
Table 5.19	Verification of displacements for sample: 7-Iso P.....	124
Table 5.20	Verification of displacements for sample: 7-Iso S.....	125
Table 5.21	Bending moment ( $M_{21}$ ) vs Isotropic soil modulus of elasticity $E_H$ for $\nu = 0.25$ .....	127
Table 5.22	Bending moment ( $M_{32}$ ) vs Isotropic soil modulus of elasticity $E_H$ for $\nu = 0.25$ .....	127
Table 5.23	Bending moment ( $M_{34}$ ) vs Isotropic soil modulus of elasticity $E_H$ for $\nu = 0.25$ .....	127
Table 6.1	Variation of bending moments $M_{21}$ , $M_{32}$ , $M_{34}$ for $E_s = 20$ MPa, and corresponding items ( $\nu_{0.25} - \nu_{0.43}$ ) in percentage.....	131
Table 6.2	Variation of bending moments $M_{21}$ , $M_{32}$ , $M_{34}$ for $E_s = 50$ MPa, and corresponding items (Tables 6.2 - Table 6.1) in percentage .....	131
Table 6.3	Variation of bending moments $M_{21}$ , $M_{32}$ , $M_{34}$ for $E_s = 70$ MPa, and corresponding items (Tables 6.3 - Table 6.1) in percentage .....	131
Table 6.4	Variation of nodal displacements/rotation $\Delta_{x1}$ , $\Delta_{y1}$ , $\Phi_{z1}$ for $E_s = 20$ MPa, and corresponding items ( $\nu_{0.25} - \nu_{0.43}$ ) in percentage.....	132
Table 6.5	Variation of nodal displacements/rotation $\Delta_{x1}$ , $\Delta_{y1}$ , $\Phi_{z1}$ for $E_s = 50$ MPa, and corresponding items (Tables 6.5 - Table 6.4) in percentage .....	132
Table 6.6	Variation of nodal displacements/rotation $\Delta_{x1}$ , $\Delta_{y1}$ , $\Phi_{z1}$ for $E_s = 70$ MPa, and corresponding items (Tables 6.6 - Table 6.4) in percentage .....	132
Table 6.7	Variation of nodal displacements/rotation $\Delta_{x3}$ , $\Delta_{y3}$ , $\Phi_{z3}$ for $E_s = 20$ MPa, and corresponding items ( $\nu_{0.25} - \nu_{0.43}$ ) in percentage.....	133
Table 6.8	Variation of nodal displacements/rotation $\Delta_{x3}$ , $\Delta_{y3}$ , $\Phi_{z3}$ for $E_s = 50$ MPa, and corresponding items (Tables 6.8 - Table 6.7) in percentage .....	133
Table 6.9	Variation of nodal displacements/rotation $\Delta_{x3}$ , $\Delta_{y3}$ , $\Phi_{z3}$ for $E_s = 70$ MPa, and corresponding items (Tables 6.9 - Table 6.7) in percentage .....	133
Table 6.10	Variation of bending moments $M_{21}$ , $M_{32}$ , $M_{34}$ for $E_v = 26$ MPa and $E_v = 35$ MPa in a soft to medium soil, corresponding ( $E_{v35} - E_{v26}$ ) in percentage .....	138
Table 6.11	Variation of bending moments $M_{21}$ , $M_{32}$ , $M_{34}$ for $E_v = 75$ MPa and $E_v = 84$ MPa in a medium to hard soil, corresponding (Table 6.11 - Table 6.10) in percentage .....	138
Table 6.12	Nodal displacements/rotation $\Delta_{x1}$ , $\Delta_{y1}$ , $\Phi_{z1}$ for $E_s = 26$ MPa, and 35 MPa in a soft to medium soil, corresponding ( $E_{v35} - E_{v26}$ ) in percentage.....	139
Table 6.13	Nodal displacements/rotation $\Delta_{x1}$ , $\Delta_{y1}$ , $\Phi_{z1}$ for $E_s = 75$ MPa, and 84 MPa in a medium to hard soil, corresponding (Table 6.13 - Table 6.12) in percentage.....	139
Table 6.14	Nodal displacements/rotation $\Delta_{x3}$ , $\Delta_{y3}$ , $\Phi_{z3}$ for $E_s = 26$ MPa, and 35 MPa in a soft to medium soil, corresponding ( $E_{v35} - E_{v26}$ ) in percentage.....	140
Table 6.15	Nodal displacements/rotation $\Delta_{x3}$ , $\Delta_{y3}$ , $\Phi_{z3}$ for $E_s = 75$ MPa, and 84 MPa in a medium to hard soil, corresponding (Table 6.15 - Table 6.14) in percentage.....	140
Table 6.16	Variation of bending moments $M_{21}$ , $M_{32}$ , $M_{34}$ versus soil shear modulus of elasticity $F_v$ .....	142
Table 6.17	Variation of nodal displacements/rotation $\Delta_{x1}$ , $\Delta_{y1}$ , $\Phi_{z1}$ versus soil shear modulus of elasticity $F_v$ .....	142
Table 6.18	Variation of nodal displacements/rotation $\Delta_{x3}$ , $\Delta_{y3}$ , $\Phi_{z3}$ versus soil shear modulus of elasticity $F_v$ .....	143
Table A1.1	Coefficient for deflection at the centre of rectangle $a \times b$ .....	1-2

Table A2.1	Presumptive values of CBR for different types of soil (Austroads, Pavement and Design 1992) .....	2-16
Table A5.1	Bending moment $M_{21}$ vs Isotropic soil modulus of elasticity $E_H$ for $\nu=0.25$ .....	5-1
Table A5.2	Bending moment $M_{32}$ vs Isotropic soil modulus of elasticity $E_H$ for $\nu=0.25$ .....	5-1
Table A5.3	Bending moment $M_{34}$ vs Isotropic soil modulus of elasticity $E_H$ for $\nu=0.25$ .....	5-1
Table A5.4.	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ ...	5-2
Table A5.5	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ ...	5-2
Table A5.6	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	5-2
Table A5.7.	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ ...	5-3
Table A5.8	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ ...	5-3
Table A5.9	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	5-3
Table A5.10	Bending moment ( $M_{21}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	5-4
Table A5.11	Bending moment ( $M_{32}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	5-4
Table A5.12	Bending moment ( $M_{34}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	5-4
Table A5.13	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ ...	5-5
Table A5.14	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ ...	5-5
Table A5.15	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	5-5
Table A5.16	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ ...	5-6
Table A5.17	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ ...	5-6
Table A5.18	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	5-6
Table A5.19	Bending moment ( $M_{21}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-7
Table A5.20	Bending moment ( $M_{32}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-7
Table A5.21	Bending moment ( $M_{34}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-7
Table A5.22	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-8
Table A5.23	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-8
Table A5.24	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ ....	5-8
Table A5.25	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-9
Table A5.26	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-9
Table A5.27	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ ....	5-9
Table A5.28	Bending moment ( $M_{21}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-10
Table A5.29	Bending moment ( $M_{32}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-10
Table A5.30	Bending moment ( $M_{34}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-10
Table A5.31	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-11
Table A5.32	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-11
Table A5.33	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ ..	5-11
Table A5.34	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-12
Table A5.35	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ .....	5-12
Table A5.36	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity of cross-anisotropic soil $E_v$ ..	5-12
Table A5.37	Bending moment ( $M_{21}$ ) vs soil modulus of elasticity of cross-anisotropic soil $F_v$ .....	5-13
Table A5.38	Bending moment ( $M_{32}$ ) vs soil modulus of elasticity of cross-anisotropic soil $F_v$ .....	5-13
Table A5.39	Bending moment ( $M_{34}$ ) vs soil modulus of elasticity of cross-anisotropic soil $F_v$ .....	5-13

Table A5.40	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv.....	5-14
Table A5.41	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv.....	5-14
Table A5.42	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv ..	5-14
Table A5.43	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv.....	5-15
Table A5.44	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv.....	5-15
Table A5.45	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv ..	5-15

# LISTING OF GRAPHS

Graph 5.1	Bending moment ( $M_{21}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	100
Graph 5.2	Bending moment ( $M_{32}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	100
Graph 5.3	Bending moment ( $M_{34}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	100
Graph 5.4	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	101
Graph 5.5	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	101
Graph 5.6	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	101
Graph 5.7	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	102
Graph 5.8	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	102
Graph 5.9	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	102
Graph 5.10	Bending moment ( $M_{21}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	103
Graph 5.11	Bending moment ( $M_{32}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	103
Graph 5.12	Bending moment ( $M_{34}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	103
Graph 5.13	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	104
Graph 5.14	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	104
Graph 5.15	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	104
Graph 5.16	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	105
Graph 5.17	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	105
Graph 5.18	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.43$ .....	105
Graph 5.19	Bending moment ( $M_{21}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	107
Graph 5.20	Bending moment ( $M_{32}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	107
Graph 5.21	Bending moment ( $M_{34}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	107
Graph 5.22	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	108
Graph 5.23	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	108
Graph 5.24	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	108
Graph 5.25	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	109
Graph 5.26	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	109
Graph 5.27	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	109
Graph 5.28	Bending moment ( $M_{21}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	110
Graph 5.29	Bending moment ( $M_{32}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	110
Graph 5.30	Bending moment ( $M_{34}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	110
Graph 5.31	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	111
Graph 5.32	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	111
Graph 5.33	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	111
Graph 5.34	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	112
Graph 5.35	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	112
Graph 5.36	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Ev .....	112
Graph 5.37	Bending moment ( $M_{21}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv .....	114

Graph 5.38	Bending moment ( $M_{32}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv .....	114
Graph 5.39	Bending moment ( $M_{34}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv .....	114
Graph 5.40	Nodal displacement ( $\Delta_{x1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv .....	115
Graph 5.41	Nodal displacement ( $\Delta_{y1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv .....	115
Graph 5.42	Nodal rotation ( $\Phi_{z1}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv .....	115
Graph 5.43	Nodal displacement ( $\Delta_{x3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv .....	116
Graph 5.44	Nodal displacement ( $\Delta_{y3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv .....	116
Graph 5.45	Nodal rotation ( $\Phi_{z3}$ ) vs soil modulus of elasticity of cross-anisotropic soil Fv .....	116
Graph 5.46	Verification of results for bending moment ( $M_{21}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	126
Graph 5.47	Verification of results for bending moment ( $M_{32}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	126
Graph 5.48	Verification of results for bending moment ( $M_{34}$ ) vs soil modulus of elasticity ( $E_H$ ) Isotropic $\nu = 0.25$ .....	126
Graph A2.1	Correlation of static cone penetrometer and CBR (Pavement Design 1992).....	2-12
Graph A2.2	Correlation of dynamic cone penetrometer and CBR (Pavement Design 1992) ...	2-13
Graph A2.3	Variation in stiffness parameters used in equation (2.79) with load repetitions (Barret and Smith (1976) ) .....	2-15



# CHAPTER ONE

## *INTRODUCTION*

*Elastostatic Interaction Analysis of Frames  
Resting on Homogeneous Elastic Half-space*

# CHAPTER 1

## INTRODUCTION

### 1.1. THESIS OUTLINE

The study described in this thesis focuses on the analysis of the interaction of elastostatic linear plane frame structures and their supporting media modeled as homogeneous elastic half-space with isotropic and cross-anisotropic properties. In the isotropic soil idealisation, Boussinesq's solutions (1885) are utilised for the application of vertical point load to the surface of elastic half-space, and Cerruti's solutions (1882) are used for the horizontal loads. The analysis is enhanced by a well-known force-displacement analytical method, termed the Direct Stiffness Method (DSM). For the case of cross-anisotropic media, the solutions of Gerrard and Harrison (1970a) and Gerrard and Wardle (1973) have been implemented to express the force-displacement relationship. These models are described in detail in the next chapter. Figure 1.1 shows a flow chart of the thesis chapters and their organisation.

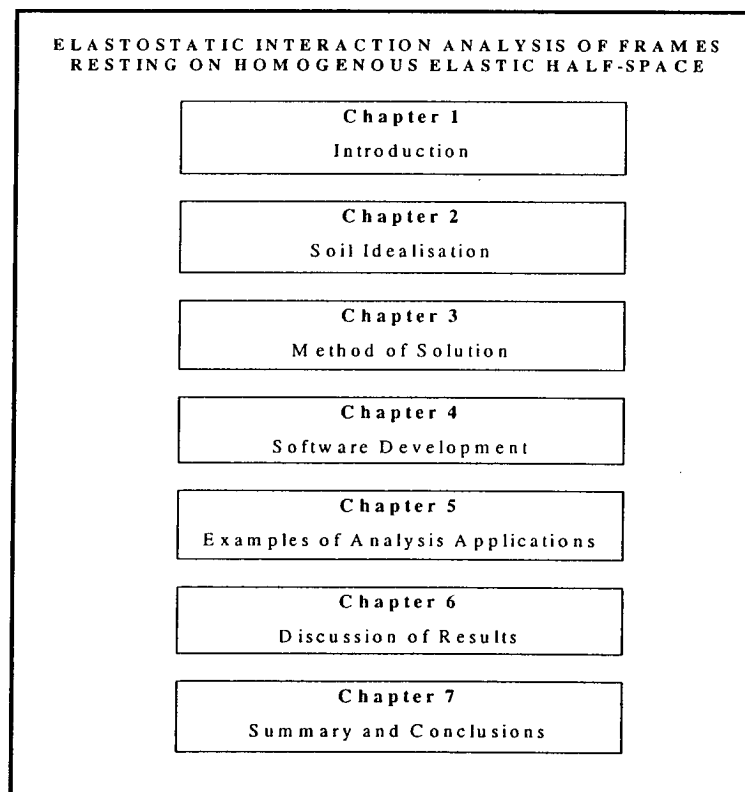


Figure 1.1 Flow chart illustrating the thesis layout.

Chapter 1, is an introductory section, which establishes the context of soil-structure interaction. The importance of the soil characteristics in the analysis is described and a summary of past work is outlined in a literature review. Different soil models are reviewed, and their important parameters are identified. The chapter concludes with a description of possible benefits that may result from the project.

Chapter 2 outlines the general idealisation of the superstructure and soil models in use (sections 2.3.2 and 2.4). The soil is represented by a half-space medium with isotropic and cross-anisotropic properties.

Chapter 3 establishes the descriptive matrices for the soil and the superstructure using the proposed analytical/ numerical models.

Chapter 4 describes the different processes and procedures that were developed and used in each step via their corresponding flow charts. These were constituted into an integrated computer package (*SASIAP*) that facilitates the analysis of soil-structure interaction for different applications. The package enables the user to extend the application of the models that were utilised for a plane frame analysis, to the third dimension for analysing the interaction caused by the neighbouring frames. The total assemble then constitutes a space frame.

Chapter 5 presents a number of soil-structure examples using different soil models. In these examples, superstructures with different support conditions are considered, and typical results obtained are tabulated for discussion in chapter 6.

In chapter 6 a discussion of the model output is carried out, and some conclusions are presented on the effects of the various parameters on model output.

In chapter 7 the final summary conclusions are presented considering both the present work and past studies. These are followed by suggestions for extension of this study and future work.

## **1.2. STATE OF THE ART: SOIL-STRUCTURE INTERACTION**

In the early years of investigating soil-structure interaction, the analyses of the structure and that of the soil were undertaken independently. In such an approach, the structural responses within a building were calculated on the assumption that the footings of the structure were rigid. The resulting reaction forces at the supports were then used by soil mechanics specialists to calculate foundation settlements based on a completely flexible structure. However, with more research in this field, new methods of analyses were developed that form the basis of a more rational approach to foundation design that integrates soil and structure (Lee & Harrison 1970; Goschy 1978; Desai et al. 1982; Masih 1985, 1993).

At present, it is widely recognised that the response of a structure is strongly dependent upon the behaviour of the soil underneath. Due to different soil support conditions, the structure can undergo various deflections in response to its external loads. These deflections are associated with soil deformation, which has two components: the short term elastic component, and the long term (or permanent) plastic (consolidation) component. This research focuses on the elastic solutions for the soil deformation.

To consider the nature of loads applied to the soil, there are two major types: static and dynamic. The response of the soil structure and the interaction of the soil and the structure vary greatly with load type. A study of all load types is beyond the scope of this thesis and only static loads are considered.

Analytical methods are accepted as one major method used to tackle soil-structure interaction problems, as these are capable of taking several factors into account. These factors include the type of foundation and the supporting soil medium, the boundary conditions and finally, the external loads. In the majority of practical cases, the analysis of soil-superstructure interaction is concerned only with the assessment of structural behaviour at working loads. Under these circumstances the magnitudes of the applied loads are relatively low and both the soil and the structural components can practically be considered as linear elastic materials. The present discussion of the interaction analysis is thus restricted to the domain of linear

elasticity. Numerous refinements are possible depending upon the soil type, which can be classified as an environment with elastic, elasto-plastic, visco-elastic or even critical state material with time-dependent properties.

The degree of cohesion and density of soil determines the responses of the supporting medium with respect to the interaction (Masih 1985). Existence of water and its quantity are other important issues that designers are required to be aware of as all engineers recognise that the presence of water can greatly alter the performance or response characteristics of the soil.

### **1.3. SIGNIFICANCE OF SOIL CONSIDERATION**

The analysis of the interaction between a structure's foundation and the supporting soil medium is of great importance to structural and geotechnical engineering. Theoretical results can provide information that can be used in both foundation and structural design. The quality of soil is important in structural design as it affects the size of the members as well as the foundation of the structure, and hence influences the economy of the structure.

The amount of research work carried out in the past, and the broad present interest in this field testify to the significance of soil characteristics in soil-structure interaction analysis. This interest has also motivated the extension of investigations into different aspects of the soil-structure interaction.

One aspect of soil characteristics is its behaviour due to external dynamic loads. Seed et al. (1975) studied dynamic soil-structure interaction with respect to the seismic design of nuclear power plants and pump stations. These researchers emphasised the significance of soil in prediction performance and the requirement for some design procedures to increase the validity of analysis techniques. Furthermore, they concluded that uncertainties in determining soil properties give rise to difficulties in accurately evaluating the characteristics of ground movements.

Meek and Wolf (1992, 1994) studied the dynamic response of the foundations of a frame, and compared the analysis of 2D- versus 3D- frame modeling of surface

foundations. They pointed out that the static response between the footings is different from that of the dynamic case. The result of the interaction in the static case may fall in an expected range. However, in the dynamic case the material properties of soil, when included in the analysis, require some alteration to accurately predict field performance.

Dieterman and Metrikine (1996) studied the dynamic properties of isotropic elastic half-space medium. Later, Metrikine and Dieterman (1997) proposed equivalent vertical stiffness for such a medium interacting with a beam taking into account the shear stress acting in the beam.

More recently, Gazetas and Mylonakis (1998) considered a variety of soil models to evaluate the soil-structure interaction. Furthermore, in the study by Stewart et al. (1999a), the analytical procedures and system identification techniques for evaluating the inertial soil-structure interaction effects on a seismic structural response were discussed. A collective examination of the empirical and predicted results from a number of sites revealed a pronounced influence of structure-to-soil stiffness ratio on inertial interaction, as well as secondary influences from the structure aspect ratio and the foundation embedment, type, shape and flexibility (Stewart et al. 1999b).

To express the significance of the "static" interaction of soil and structure, Meyerhof (1947), Francis (1954) and Chamecki (1956) developed interaction analyses for multi-storey multi-bay structures on isolated frames. These analyses were superseded by Lee's method (Lee 1969), which required only minor amendments to those previous analysis procedures. This method was named "*the fictitious member technique*", where the supporting soil was replaced by a small structural member at each column base where its axial stiffness was considered equal to the relevant stiffness of the footing-supporting soil at each column. Lee (1975) emphasised the analysis of structure based on the above technique, and considered the interaction for the major types of foundations, namely, isolated footings, piles and rafts.

In his work, Brown (1975) emphasised the significance of structure-foundation interaction. An examination of the importance of such interaction with regard to the

effect of differential displacements of column bases and the internal forces was presented, and he investigated the important factors that control the magnitude of the interaction. In his findings, several factors such as the combination of relative stiffness at which reduction in differential displacement occurs, increase in column stiffness, the number of storeys and the thickness of the beam foundation, can possibly decide when full interaction effects are of little significance and a complete analysis can be avoided.

For an economical design of a structure, it is important to consider the soil as part of the medium that is exposed to the external loads. In such a design, it is essential to study the loads transmitted to the soil and the resulting stress distributions within the soil itself. The study of the physical state of foundation materials, and overall foundation plan and design, are important in avoiding any excess settlement or soil failure. A complete interaction analysis should involve the following:

- a) The distribution and intensity of pressure between the footing and the foundation medium (contact pressure)
- b) The intensity of normal and shearing stresses at various points within the mass of the medium
- c) Any possible mechanism of soil failure underneath the foundation after considering the soil's physical characteristics as found by initial soil tests.

Over the years, various computer packages have been developed that idealise soil using finite elements and employ various assumptions. Some methods need more input data for their parameters than other approaches. In the later sections of this chapter, a review of the process of development of soil models and methods is presented.

## **1.4. LITERATURE REVIEW**

Due to its nature, the behaviour of a soil is complex under external loads. In the past decades extensive studies have been undertaken to model soil stress distribution. As indicated previously, there are two major load types that can be considered for interaction: static and dynamic and this study involves static loads only.

With static interaction analyses, there are three major categories for soil models:

- Infinitely rigid,
- Elastic, which includes single-parameter, modified single-parameter, homogeneous isotropic half-space, homogeneous cross-anisotropic half-space and non-homogeneous half-space,
- Visco-elastic/ plastic media which is not the focus of this research.

Solutions to express soil behaviour based on the assumption of a continuum can be grouped according to the classification of the half-space in question. Stress calculations can be classified from a mathematical point of view, that is, whether they lead to a closed form (for example elastic half-space) or an open form (numerical methods, for example Finite Element Method (FEM)) solution. In Figure 1.2, the shaded models are the soil idealisations that are considered in this project.

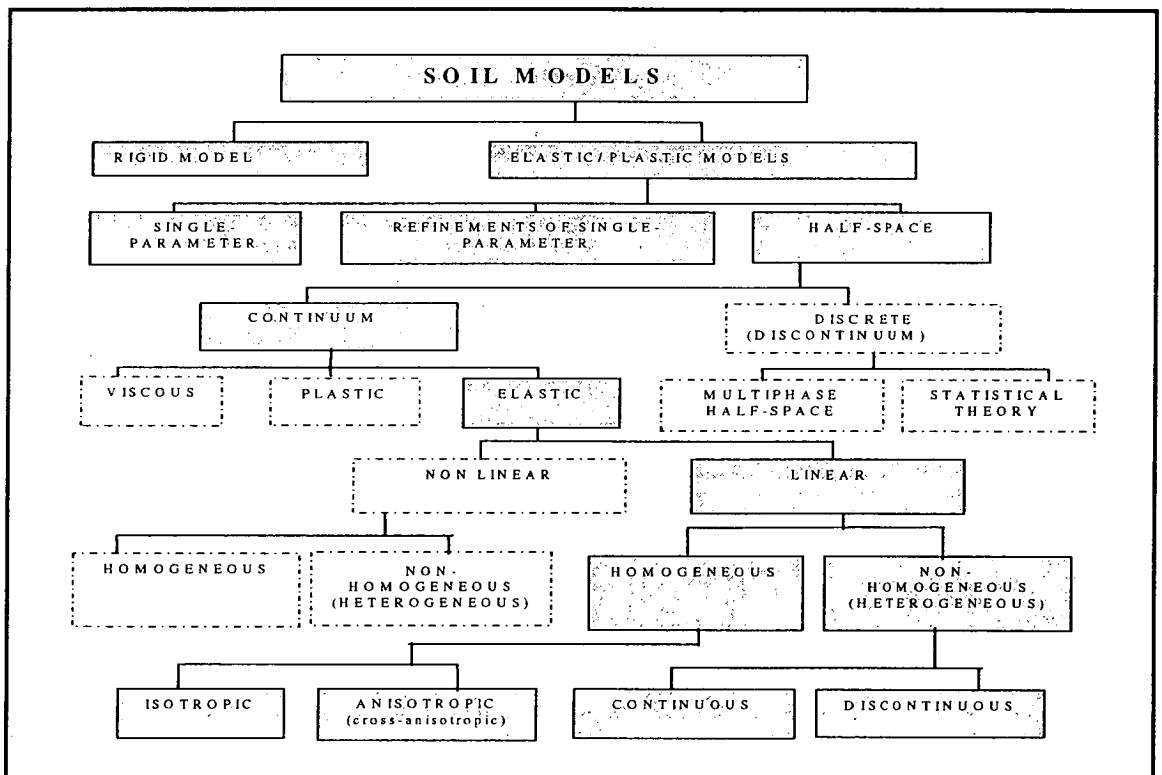


Figure 1.2 Classification of soil models and methods of stress consideration and calculation.

Initially, a single-parameter model (sub-grade modulus) is used to describe soil. This model is quite simple to use, although it can not sufficiently express the real behaviour of soil under the different types of external loads. Due to the importance of soil characteristics and behaviour, continuous efforts have been made over the years



to introduce new and improved soil models. However, these models often appeared to have some shortcomings. To rectify these deficiencies, successive investigators defined extra parameters in their idealisations as refinements to earlier models. Examples are flexural rigidity (D) and tension field (T).

As a result of later efforts in adopting a soil model, a homogeneous isotropic elastic continuum (the soil parameters are Young's modulus and Poisson's ratio) was identified. In addition, for a more complex case (for example a layered soil), a homogeneous cross-anisotropic half-space was defined in which five independent parameters are used to describe the medium.

The majority of the existing solutions using analytical or numerical techniques for isotropic or cross-anisotropic homogeneous half-space are in general tedious to utilise. With these approaches, solutions often involve differential equations of high order, which produce many answers. In these cases it is required to implement the boundary conditions to confirm acceptable results, and to simultaneously eliminate the general answers to those differential equations.

Poulos (1975a) analysed settlement of structure-foundation systems considering relative structural stiffness. Davis and Poulos (1968) and Poulos (1975b) summarised some of the more commonly used methods [such as the Conventional One-dimensional method, Skempton and Bjerrum's (1957) method, the Effective Stress Path method, the Elastic method, Cambridge approach and Finite Element Method (FEM)], to calculate the settlement of isolated foundations.

Many studies have been carried out (Burland and Burbidge 1985, Bowles 1987, Papadopoulos 1992, Wahls 1994, Montrasio and Nova 1997, and Maugeri et. al 1998) on the elastic settlement of sand deposits. Burland and Burbidge (1985) examined an extensive number of cases to obtain a consistent picture to evaluate stiffness, namely operational stiffness for non-cohesive soils for a continuum mechanics approach. Operational stiffness is defined as an equivalent stiffness for the non-cohesive soil medium and is obtained by using elastic equations. Burland and Burbidge concluded that due to the variability of soil, elastic settlement could vary from the expected values by a factor of 1.5. Berardi and Lancellotta (1991)

confirmed the application of operational stiffness for evaluation of settlements in non-cohesive soils.

Cheung and Zienkiewicz (1965) found that the Winkler type spring approximation introduced to avoid mathematical difficulties need no longer be used where continuous foundations are present, and no special treatment of holes, corners or other irregularities in the foundation plate was necessary (refer to Appendix A1). Therefore, practical cases such as variable thickness foundation rafts or other shapes of foundations were capable of rapid solution. More recently Montrasio and Nova (1997) studied the effect of shape of foundation on the settlement of shallow foundations where they employed a number of mathematical methods. They found that experimental evidence is generally well matched by theory of elasticity. It was shown that only two out of nine parameters that characterised their model varied significantly with the shape of the foundation. They concluded that embedment has a small influence on the value of additional two parameters and the other five remain constant.

There are several soil models used in soil-structure interaction analysis and some of these are briefly reviewed in this chapter. This study however, focuses on soil medium with linear homogenous isotropic and cross-anisotropic properties. For the former, the solutions of Boussinesq (1885), Cerruti (1882) and Mindlin (1936) have been used and for the latter, those presented by Gerrard and Harrison (1970a) and Gerrard and Wardle (1973) have been implemented. Of these approaches, Mindlin's solution applies to the interior of the half-space while the remaining approaches are applicable to the surface of the medium.

#### **1.4.1. Rigid Model**

In the early days of structural analysis, soil was considered as a rigid medium that experienced no deflection due to applied loads. Therefore, any flexibility of the foundation of the superstructure was ignored, and it was assumed that members of the superstructure, on the basis of their stiffness, would withstand external loads. Eliminating soil behaviour leads to an inadequate understanding of any interaction between soil and superstructure and inefficient structural design. As both theory and

practice imply, any foundation displacement affects the internal forces of the superstructure. Therefore, this early idealisation of soil was unrealistic and encouraged research into utilising the elastic behaviour of soil in the structural analyses of the building.

### 1.4.2. Winkler Model and Its Applications

The Winkler springs or Winkler model (1867) was the first elastic idealisation of soil. According to this theory, soil is assumed to be a series of independent springs, all with a constant stiffness, that react against vertical forces transferred from the structure. These vertical reaction forces are considered directly proportional to the local vertical displacement of the foundation. This model is a single-parameter soil idealisation used to describe the stresses produced in the soil beneath the foundation. The relationship is defined by a parameter ( $k_s$ ) known as the modulus of sub-grade reaction, and for a beam resting on a medium idealised by such a model (two-dimensional application) the stress is given by

$$p(x) = k_s w(x) \quad (1.1a)$$

where  $k_s$  is the modulus of foundation or sub-grade reaction, with its dimension being in pressure per length of vertical displacement in the range of 5 - 50  $MN / m^3$ . The function  $w(x)$  is the vertical deflection (i.e.  $z$  direction, perpendicular to the beam) beneath the foundation (Figure 1.3).

The general form of the Winkler model in a three-dimensional application (a plate in  $x - y$  plane) is described as

$$p(x, y) = k_s w(x, y) \quad (1.1b)$$

where  $p(x, y)$  is the intensity of reaction of the soil, and  $w(x, y)$  is vertical deflection (in the  $z$  direction, perpendicular to the plate) beneath the foundation.

In Figure 1.3, the sign convention and the applied load are presented for both a beam and a plate. Whilst Figure 1.4 illustrates the behaviour of a Winkler foundation.

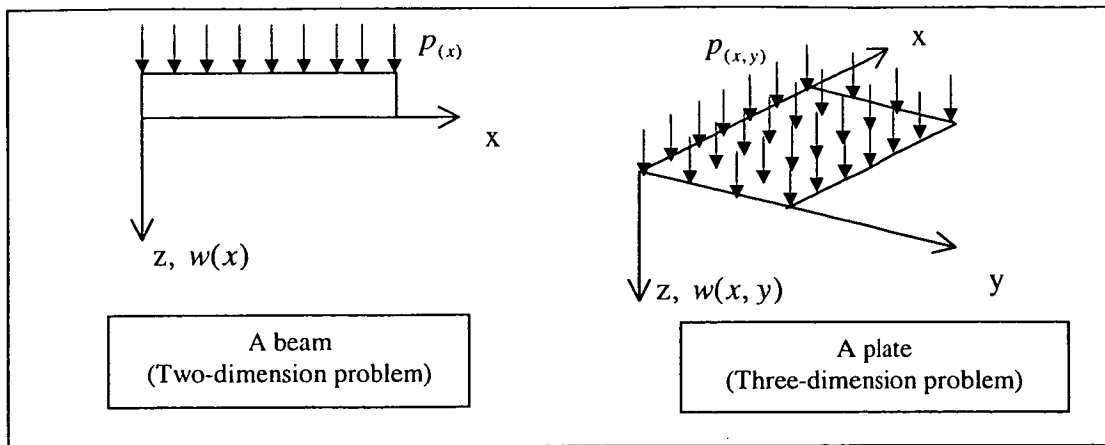


Figure 1.3 Illustration of sign convention for a beam and a plate foundation resting on Winkler model.

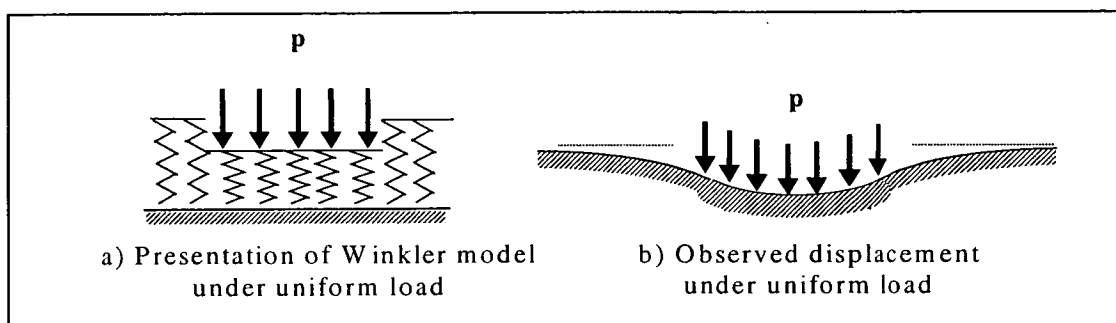


Figure 1.4 Illustration of soil deformations: (a) under external load for Winkler model, and (b) the observed deflections on site in the real situation.

From Figure 1.4a, it can be observed that in a Winkler medium, the displacement under the loaded area is constant when it is subjected to a uniform external load, whilst the displacement outside the loaded zone is zero. The observed field behaviour of soil under a uniform load is as shown in Figure 1.4b. This difference between the behaviour of the real soil and that of a Winkler soil is one of the major shortcomings of the Winkler model. For this reason, several researchers have suggested refinements to the Winkler model to improve its application as a soil idealisation.

The applications of the Winkler model and its refinements using different methods of analysis and calculation have been the topic of much research and investigation. Some of this research related to beam and plate structures is presented below. Hetenyi (1946), and Timoshenko and Woinowsky-Krieger (1959) investigated the problem of beams and plates on the Winkler model. Although this model is simple to

use there are some inherent disadvantages. As indicated earlier, it is the lack of continuity in the supporting medium in the model that is physically unrealistic.

Many researchers, (Lee and Brown 1972 and Selvadurai 1979) addressed the model's limitations and inherent difficulties in solving soil-structure interaction problems using various continuum models. Vlasov and Leontiev (1966) attempted to overcome the shortcomings of the model by incorporating foundation flexural rigidity as a second parameter in soil idealisation for the analysis of a rigid circular foundation under both uniform and non-uniform vertical load.

Schleicher (1926) and Conway (1955) studied the axi-symmetrical loading of a circular foundation using the Winkler model, and Hetenyi (1946 and 1950) suggested an interaction between independent spring elements and a beam (2D problem) or plate (3D problem), assuming bending in the beam or plate.

The Winkler model was further investigated and developed using different methods of analysis and different numerical techniques for the analysis. A number of researchers in the past, Lee and Brown (1972), Selvadurai (1979), and more recently, Melerski (1992, 1995a, 1995b) used finite element and finite difference techniques in analysing soil-structure interaction via Winkler springs.

As indicated earlier, the Winkler model is a single-parameter soil idealisation. Over the years, extra parameters such as tension field or shear interactions between the spring elements, and flexural rigidity for foundation plate were incorporated into the analysis to rectify the shortcomings of this soil model (Filonenko-Borodich 1940, Hetenyi 1946 and Pasternak 1954). Further discussion on Winkler refinements can be found in Kerr (1964).

Hemsley (1988) considered the flexure of an infinite plate on Winkler springs and a half-space model. In this study, a plain strain energy theory based on Fourier transforms was applied to link the elastostatic flexure of a thin infinite plate founded both in Winkler springs and a homogeneous half-space. Simple closed-form solutions were derived for a plate on springs under symmetric and anti-symmetric loadings, where numerical results were compared with those obtained for an infinite

plate in frictionless and continuous contact with a homogenous half-space. The significance of the homogeneous half-space model is discussed in the next section.

### **1.4.3. Homogeneous Isotropic Elastic Half-space Model**

One of the frequently used models for soil idealisation is a homogeneous isotropic elastic half-space. There are two parameters that are utilised in this model, modulus of elasticity ( $E$ ), and Poisson's ratio ( $\nu$ ). The first parameter associates the deformation in the soil and the stresses applied to it when these are in the same direction. Vertical deformation in the medium is related to the stress applied to the medium in the horizontal direction and vice versa. This relationship is expressed by Poisson's ratio.

Boussinesq (1885) proposed soil to be an elastic homogeneous half-space medium of isotropic material. He solved the problem of a vertical point load acting on the surface of the medium and gave solutions for the stresses and displacements at any point in the medium. For similar media Cerruti (1882), analysed the problem of a horizontal point load acting on the surface and developed solutions for the stresses and displacements at any point within the medium. The above two researchers were at the frontier in solving the problems of elastic isotropic half-space at that time.

Based on the fundamental results from the theory of elasticity known as the Kelvin solution (1848), further studies were undertaken by Love (1927) and Timoshenko (1934). Solutions to the problem of stress distribution for a number of cases have been obtained using the Kelvin solution.

For application of loads within a homogenous elastic half-space, Melan (1932) developed solutions to evaluate stresses and strains for a two-dimensional application. Mindlin (1936) provided similar expressions for a three-dimensional application. These expressions were similar in nature to Boussinesq's and Cerruti's findings for surface loads.

Vogt (1925) followed by the U.S. Bureau of Reclamation (USBR) (1956) utilised Boussinesq's and Cerruti's solutions for the study of uniformly distributed load and

moment on an elastic half-space. Lee and Brown (1972) performed a comparative analysis of the structure-foundation interaction between Winkler's model and linear elastic models in the form of the Boussinesq model.

In further studies, Hain and Lee (1980) examined the influence of interaction between a 3D structure and its footing founded on isotropic elastic half-space. They concluded that the interaction behaviour of a three-dimensional frame and a raft foundation can be predicted if the relative stiffness of structures ( $K_s$ ) to the rafts ( $K_R$ ) is in the range:  $0.1 < K_s / K_R < 10.0$  then redistribution of column loads largely occurs. They also found that an increase in differential settlements, maximum negative bending moment, and reduction in maximum positive bending moment occur. This is due to increased interaction caused by an increase in the number of bays in the structure.

In the past few years, there have been a number of studies undertaken by several researchers in relation with soil-structure interaction. Huang and Tatsuoka (1988), Khing et al. (1992) and Takemura et al. (1992) considered improvements for analysing and predicting behaviour for footings placed in sandy ground. Delgado and Faria (1994) investigated soil-structure interaction for dam-foundation-reservoir interaction to obtain assurance of dam safety and performance due to earthquakes. In this work numerical modeling aspects of foundations were described.

Huang and Menq (1997) analysed deep footing and width effects in the elastic half-space medium by considering the results of 105 models tested in their investigation. More recently, in deep-foundations studies, Shen et al. (1997 and 1999) applied a variational approach to analyse vertically loaded pile groups. They determined that displacement at the base of the piles by their method conformed to that obtained using Mindlin's (1936) solution.

Schleicher (1926), Steinbrenner (1934), Florin (1961) and Giroud (1968, 1970 and 1973) utilised Cerruti's and Boussinesq's solutions to evaluate stresses and displacements in soil due to a distributed load, whereas Giroud (1968, 1970 and 1972) evaluated the same unknowns by introducing influence factors.

Holl (1940), Fadum (1948), Scott (1963) and Harr (1966) extended the study on homogeneous isotropic elastic half-space over different shapes of vertically loaded foundations and developed solutions for the stresses beneath the foundations. Ahlvin and Ulery (1962), and Poulos and Davis (1974) carried out extensive studies on the elastic solutions for isotropic elastic half-space. Recently, Bull (1994) presented a study on the various numerical analyses and modeling methods including Finite Element Method (FEM) employed in soil-structure interaction and concluded that small differences in loaded area and shape applied pressure did not significantly effect the stress, except at the surface under the load.

Newmark (1935, 1942 and 1947) proposed solutions to the problem of stresses and displacements using a graphical approach. Foster and Ahlvin (1954) and Ahlvin and Ulery (1962) used tables to present their solutions for a wide range of Poisson's ratio. Poulos (1967) utilised the sector method and curves to solve the same problem.

Meyerhof (1947), Francis (1954) and Chamecki (1956) developed several methods. Meanwhile, Lee (1975), Lee and Hain (1974) and Lee and Valliappan (1974) conducted a series of studies, which were concentrated on isolated foundations and proposed a "fictitious element" method.

Several studies have been conducted to determine the stresses and displacements in soil. Borowicka (1936), Gorbunov Passadov (1949), Ishkova (1957), and Gorbunov Passadov et al. (1984) considered the stiffness of superstructure and applied power series methods in their solutions. Brown (1969) introduced plate stiffness in his solutions for analysing the stress and displacement generated in soils under foundations.

Recently, a range of approaches and analyses in soil structure interaction were taking the interest of the researchers and scholars. Brown (1969), Gorbunov Passadov et al. (1984), Milovic and Djogo (1991), and more recently, Melerski (1995a, 1995b) considered foundations with various rigidities. In these studies, analytical and numerical methods were utilised. In the next chapter, selected soil models are presented and those applicable to this research are elaborated.



#### **1.4.4. Homogeneous Cross-anisotropic Elastic Half-space**

##### **Model**

A more complex situation exists in a homogeneous medium when Young's modulus and Poisson's ratio vary in different directions, that is cross-anisotropic behaviour is encountered. In a soil the homogeneous cross-anisotropic model is typically associated with a layered (sedimentary) soil medium.

In 1900, Michell derived solutions for stress-strain relations due to the application of a vertical load to a medium with cross-anisotropic properties, and was followed by Koning (1957), Anon (1960), Lekhnitskii (1963) and Barden (1963). The Suklje (1963) solution confirmed soil properties for an anisotropic case, and investigated the elasticity of the medium by conducting triaxial tests. Dooley (1964) commented on Barden's solutions, indicating that Barden made the implicit assumption that the soil shear modulus for a pair of axes inclined at 45 degrees to the x-z axes was the same as that in the direction of x and z axes. Dooley (1964) concluded that the validity of that solution was limited. Urena et al. (1966) considered the case of a vertical point load in an infinite media with horizontal planes of discontinuity (cross-anisotropic media), and obtained the distribution of stresses and displacements in the modelled soil. They extended this approach to a load acting along a straight line, and also investigated the case of a uniform tangential as well as vertical loads acting on the surface of a half-space.

Hooper (1975) considered the effect of transverse isotropy on the surface settlement of a homogeneous elastic half-space underlain by a rigid frictionless layer resulting from both frictionless and adhesive axi-symmetric surface loading applied over a circular area. He concluded that particular emphasis must be applied to the sensitivity of the calculated settlements as a function of the assumed values of the elastic constants.

Gerrard (1967) investigated vertical uniform loading on a strip foundation and obtained a number of solutions for stresses and strains generated in the soil. Lempriere (1968) studied an anisotropic medium through the strain energy method using a valid range for Poisson's ratio, and Pickering (1970) investigated the

parameters for this type of medium that Love (1892) had earlier defined with five independent parameters. In later studies, Gerrard and Harrison (1970a and 1970b) developed solutions to determine stresses and displacements for circular loads on this type of medium. These solutions were based on the application of integral transform techniques and dual integral equation techniques to elasticity problems (Sneddon 1951, Tranter 1966). Moreover, Gerrard and Wardle (1973) presented solutions for the application of point loads based on the general form of the solutions that were presented by Sneddon (1951) and Tranter (1966). The solutions presented by Gerrard and Harrison (1970a) and Gerrard and Wardle (1973) are relevant to this project and are utilised in this work (Chapters 3 and 4).

The Finite Element Method (FEM) has also been utilised to obtain the soil displacements caused by the application of surface loads. Many researchers, Cheung and Zienkiewicz (1965), Wardle and Fraser (1974), and Chandrashekhara and Antony (1993, 1996) used this technique in their analyses. Chandrashekhara and Antony (1993, 1996) found that the rigidities of frame structure with respect to soil affects the contact pressure and bending stresses in the frame. The Winkler model appeared to be inadequate for the layered system considered, while the equivalent model appeared to be adequate for contact pressure distribution as well as for determining bending stresses in the frame but unsatisfactory for the determination of surface vertical displacement.

More recently, Antony and Chandrashekhara (1997) evaluated the contact stresses for cross-anisotropic elastic foundation, and Pires and Higgins (1998) proposed a spring model for linear soil-structure interaction, and used this model to investigate the behaviour of the foundations.

#### **1.4.5. Non-homogeneous Isotropic Elastic Half-space Model**

More complex models for soil are needed when the soil is assumed to be non-homogeneous. Holl (1940) developed a general form of Boussinesq's classical equations on the basis of Griffith's (1929) and Frohlich's (1934) solutions for the problem of vertical point load on the surface of a non-homogeneous half-space.

Gibson (1967) and Brown and Gibson (1973) in their studies of non-homogeneous soils proposed the soil to be isotropic with a shear elasticity modulus linearly increasing with depth ( $G(z) = G(0) + m z$ ). This model, known as Gibson's model, formed the basis of ongoing work (Gibson and Kalsi 1974, Gibson and Sills 1975). Carrier and Christian (1973) later employed a Finite Element Method (FEM) for the study of flexible circular foundations. More recently, Antony (1994), Chandrashekhara and Antony (1996) performed the interaction analysis of footings resting on a non-homogeneous elastic medium and combined analytical and finite element methods in their studies.

Hain and Lee (1980) also investigated the behaviour of a structure resting on a perfectly elastic medium with the modulus increasing linearly with depth. Dempsey and Li (1995) considered rectangular and strip footings (both rigid and flexible) in full contact with soil. They used Gibson's model in conjunction with numerical integration. In this study, the fundamental solution of the non-homogeneous half-space was separated into the primary solution associated with the homogeneous half-space and a function corresponding to the non-homogeneity of the half-space. The latter function was then approximated by an analytical expression that eliminated most of the numerical integrals.

In a very recent, available study on non-homogeneous soil models, (Hu et al. 1999) considered circular foundations and applied numerical approaches and experimental results to successfully estimate settlement of offshore structures.

Non-homogeneous isotropic elastic half-space soil model is quite complex and the existing solutions do not fully describe a non-homogeneous medium. This deficiency raised the need for including the variation of soil parameters in certain directions within the soil mass. Further explanation is provided in the next chapter on this aspect.

Despite all the work that has been done to date it has been concluded that, more research needs to be carried out to obtain a simple and acceptably accurate framework to assess soil-structure interaction during the design phase of engineering projects.

## **1.5. REQUIREMENTS FOR SOIL-STRUCTURE ANALYSIS**

An obvious and important part in a soil-structure interaction analysis is to use a soil model that is able to describe the force-displacement relationship throughout the media due to an external load from the superstructure. The model, depending upon the soil, could be of a single-parameter, two-parameter, homogeneous or non-homogeneous, isotropic or cross-anisotropic type.

An equally important aspect is to utilise an appropriate numerical or analytical technique to "link" the superstructure with its supporting medium. By integrating these two steps, the interaction can be assessed, and the superstructure reaction as well as individual member internal forces can be obtained.

## **1.6. SCOPE OF THIS THESIS**

Framed structures are a fundamental methodology widely used in the construction industry for residential, commercial and industrial purposes. These structures are simple in design and relatively easy to install. Prefabricated structures in particular are constructed in a short period of time within a quality-controlled environment, and provide cost-effective options. In a construction site with limited space, prefabricated structures can be stored close to the site and easily transferred for quick installation. This makes such structures very popular and frequently used. The construction materials are usually steel, timber, concrete or composite. The structural members can be designed using a working stress or ultimate stress approach. The method used to analyse such structure is based on linear static analysis, in which the internal stresses remain within the linear elastic zone. Since structures are supported by the soil beneath them, the study, idealisation and characteristics of these media are important in this research.

By making use of simple yet practical analytical soil models coupled with a powerful numerical approach (such as a direct stiffness method), structures may be more easily and quickly designed. Using such models, the deformations of the superstructure and soil can be evaluated simultaneously.

The author (Izadnegahdar 1997) has prepared a computer package (Soil And Structure Interaction Analysis Package, SASIAP ver.1.0) in Turbo Pascal (7.0) language based on the needs that have been mentioned earlier for soil-structure interaction analysis. The package requires structural information for members and for the supporting medium: Poisson's ratio, Young's modulus and the width of footing. The package requires minimal input data to produce the solution to the problem. This package analyses frames, evaluates the internal forces, nodal displacements and the reactions of superstructures due to the soil-structure interaction with respect to plane frame structures that are founded on homogeneous elastic half-space with either isotropic or cross-anisotropic behaviour. The combination of such (typical) frames in parallel, (3D) and the interactions of neighbouring footings (2D) are also considered.

## **1.7. POSSIBLE BENEFITS OF THE RESEARCH**

The research has importance to both researchers and design engineers in that it provides them with a new, improved and innovative approach to the analysis and design of framed structures.

As mentioned in the literature review, solutions to the early soil models such as Winkler's model were simple to apply but did not consider a more realistic representation of soil behaviour. Some attempts, such as introducing extra parameters, were made to reduce these deficiencies but often resulted in the formulation of overly complex formulae that required the use of numerical form or power series.

Utilisation of the later soil models such as by Boussinesq (1885), Cerruti (1882), Mindlin (1936), Gerrard and Harrison (1970a) and Gerrard and Wardle (1973) gave more rational results. The author has proposed a method of utilising an analytical/numerical approach, namely the Direct Stiffness Method for analysis, based on considering the soil models with either homogeneous isotropic or cross-anisotropic properties. This approach is used to evaluate the nodal displacements of the system. The application of these soil models using analytical approaches leads to better solutions with respect to the issue of interaction between a building's footings. This is due to the simplicity and minimum input data requirement, and the

modularity of approach that could be extended to 3D space frames. These factors provide an advantage over the existing software packages that are either overly complex or require a large amount of input data.

# CHAPTER

# TWO

*SOIL IDEALISATION*

*Elastostatic Interaction Analysis of Frames  
Resting on Homogeneous Elastic Half-space*

# **CHAPTER 2**

## **SOIL IDEALISATION**

### **2.1 INTRODUCTION**

As identified in Chapter 1, the study of soil-structure interaction can be described as being composed of two components. These are superstructure modeling and soil idealisation. In this chapter some of the soil idealisation approaches are discussed.

Following the introduction to some important soil models, the chapter focuses on the force-displacement relationships suggested by various approaches for soil idealisations. These models are single-parameter soil model (Winkler springs), two-parameter soil models (Refinement of Winkler foundation, and homogeneous isotropic elastic half-space), cross-anisotropic and non-homogeneous elastic half-space.

### **2.2 SINGLE-PARAMETER SOIL MODEL**

The single-parameter soil model known as Winkler springs is initially discussed (see Chapter 1). As mentioned in the previous chapter, this model inherits some deficits such as non-continuity in the stiffness and deflection of soil medium. In the work carried out and described here, the structural parameters of flexural rigidity or shear modulus of the foundation, have been incorporated to minimise the shortcomings associated with obtaining force-displacement relations.

Hetenyi (1946, see Kerr 1964) considered Winkler springs to determine the vertical displacements of beam and plate foundations due to surface loads. In his approach, flexural rigidity (Equation 2.1b) of these structure elements was determined. It was assumed that the deformation associated with an elastic beam (in the one-dimensional case) or a plate (in the two-dimensional case) on the medium was restricted to bending only. By applying the Winkler model, the interaction between the independent spring elements is recognised, and the relation between load



$p(x, y)$ , contact stress  $q(x, y) = k_s w(x, y)$  and deflection  $w(x, y)$  for the two-dimensional case for rectangular plates can be expressed as

$$p(x, y) = k_s w(x, y) + D_p \nabla^4 w(x, y) \quad (2.1a)$$

where  $D_p$  is flexural rigidity of the plate foundation, and is given

$$D_p = \frac{E_p h_p^3}{12 (1 - \nu_p)^2} \quad (2.1b)$$

where  $E_p$ ,  $\nu_p$  and  $h_p$  are the plate's modulus of elasticity, Poisson's ratio and the height respectively.  $\nabla^2$  is Laplace operator in  $x$  and  $y$  directions and  $\nabla^4$  in rectangular coordinates, can be written as follows

$$\nabla^4 = \nabla^2 \nabla^2 = \frac{\partial^4}{\partial x^4} + 2 \frac{\partial^4}{\partial x^2 \partial y^2} + \frac{\partial^4}{\partial y^4} \quad (2.1c)$$

Several authors, Borowicka (1936), Gorbunov-Possadov (1949), Gorbunov-Possadov et al. (1984) and Ishkova (1957) used power series in their procedures to solve differential Equation 2.1a.

A case of an axi-symmetric circular plate resting on a single-parameter elastic medium has been the topic of a number of studies, and considering the plate's flexural rigidity the analysis of the plate problem involves the solution of the following differential equation

$$p(r) = q(r) + D_r \nabla^4 w(r) \quad (2.2a)$$

where  $p(r)$ ,  $q(r)$  and  $D_r$  are the load, contact stress and the flexural rigidity of the circular plate respectively.  $\nabla^4$  for a circular plate is defined in equation 2.2b.

$$\nabla^4 = \nabla^2 \nabla^2 = \frac{d^2}{dr^2} + \frac{1}{r} \frac{d}{dr} \quad (2.2b)$$

Several solutions for Equation 2.2a were provided by Schleicher (1926), Hetenyi (1946) and Conway (1955) (see Milovic 1992). They suggested that by summing the particular integration which depends on the form of  $p(r)$  a solution for the homogeneous equation in the following form can be obtained

$$w(r) = A_1 J_0\left(\pm \frac{r}{l} \sqrt{i}\right) + A_2 Y_0\left(\pm \frac{r}{l} \sqrt{i}\right) + A_3 J_0\left(\pm \frac{r}{l} \sqrt{-i}\right) + A_4 Y_0\left(\pm \frac{r}{l} \sqrt{-i}\right) \quad (2.2c)$$

In this equation,  $J_0$  and  $Y_0$  are zero order Bessel functions, and  $A_1, \dots, A_4$  are arbitrary constants and  $i = \sqrt{-1}$ .

Vlasov and Leontiev (1960, see Milovic 1992) considered plate rigidity and transverse shear. This results the following differential equation.

$$D_p \nabla^4 w(x, y) - G_p \nabla^2 w(x, y) + k_s w(x, y) = p(x, y) \quad (2.3a)$$

The solutions to the above differential equation give the surface deflection  $w(x, y)$  that includes the sum of a particular integral, which depends on the form of soil reaction  $q(x, y)$  and the solution for the homogeneous equation.  $\nabla^4$  is defined in equation 2.1c.

Vlasov and Leontiev (1960) also investigated the displacement of a circular plate resting on an elastic medium. In this study, the medium was a continuum and the solution to the differential equations was given in the form of Bessel and Hankel functions as follows

$$w(r) = B_1 J_0\left(\frac{r}{l} \sqrt{a}\right) + B_2 H_0^{(1)}\left(\frac{r}{l} \sqrt{a}\right) + B_3 J_0\left(\frac{r}{l} \frac{1}{\sqrt{a}}\right) + B_4 H_0^{(2)}\left(\frac{r}{l} \frac{1}{\sqrt{a}}\right) \quad (2.3b)$$

where  $B_1, \dots, B_4$  are arbitrary constants and  $H_0^{(n)}$  is the zero order Hankel function of the  $n^{\text{th}}$  kind.

## 2.3 TWO-PARAMETER SOIL MODELS

Two-parameter soil models are discussed below. In these idealisations the following assumptions are made:

- a) the soil is a weightless medium
- b) the medium is elastic, homogeneous, semi-infinite and isotropic, and Hooke's law applies
- c) the soil volume change is negligible
- d) there is no initial stress in the soil before load application
- e) stress continuity applies to the soil
- f) the foundation remains in contact with the soil

### 2.3.1 Refinements to Winkler Model

Several attempts have been made to consider additional parameters when using the Winkler springs for soil idealisation. Amongst these studies, Filonenko-Borodich (1940, see Kerr 1964) assumed some degree of interaction between the spring elements, where the top ends of these springs were connected to a stretched elastic membrane which was subjected to a constant tension field  $T_s$ . In such a case the force-displacement relationship for a loaded plate was described by

$$p(x, y) = q(x, y) - T_s \nabla^2 w(x, y) \quad (2.4a)$$

where  $\nabla^2$  is the Laplace operator in x and y directions,  $p(x, y)$  is the load that is applied perpendicular to the plate, and  $q(x, y)$  is the contact stress. In the Winkler model this stress is defined as  $q(x, y) = k_s w(x, y)$  (refer to Figure 1.3 for the sign convention). From Equation 2.4a, the interaction of the spring elements is characterised by the intensity of the tension field  $T_s$  in the membrane.

Another modification of the Winkler model was by Pasternak (1954, see Kerr 1964). In this approach, shear interaction between the spring elements was assumed. This was achieved by connecting the ends of the springs to a fictitious beam or a plate consisting of incompressible vertical elements that deformed only by transverse shear. It is also assumed that the medium is homogeneous and isotropic in the x-y

plane ( $G_x = G_y = G_s$ ). The relationship between the pressure and the deflection of the foundation is determined from

$$p(x, y) = k w(x, y) - G_s \nabla^2 w(x, y) \quad (2.4b)$$

where  $G_s$  is the shear modulus for the soil medium.

The generalised foundation approach is another soil idealisation where it is assumed in addition to the Winkler condition, that at each point the load pressure ( $p$ ) is proportional to the soil deflection ( $w$ ), and that the moment in the foundation beam is proportional to the angle of rotation. This model is described by:

$$p = k w \quad (2.5a)$$

and

$$m_n = k_1 \frac{dw}{dn} \quad (2.5b)$$

where  $n$  is any direction at a point in the plane of the foundation surface,  $k$  and  $k_1$  are the corresponding proportionality factors.

### 2.3.2 Homogeneous Isotropic Elastic Half-space

This section focuses on representing the soil as medium with homogeneous isotropic properties due to its broad applications in design and analysis. A half-space is typically described as a continuum elastic body having infinite horizontal dimensions and infinite depth below the horizontal plane surface.

A planar frame exposed to external loads in  $x-z$  plane and supported by an elastic half-space medium is initially considered. The structure is then replaced by a number of reactions at the supporting points which include vertical forces ( $P_z$ ), horizontal forces ( $Q_x$ ) and horizontal moments ( $M_y$ ) about the  $y$  axis. The following sections focus on the force-displacement generated in this medium due to the above reactions.

In the case of a vertical load, the axis of elastic symmetry is vertical. In this soil idealisation, the two required soil parameters are Young's modulus or modulus of elasticity ( $E_s$ ), and Poisson's ratio ( $\nu_s$ ). Boussinesq (1885), see Poulos and Davis (1974), investigated the application of a point load  $P$  normal to the surface of a homogeneous isotropic elastic half-space. The solutions to determine the stresses and displacements were obtained in cylindrical and Cartesian coordinates. Boussinesq's approach is shown in Figure 2.1.

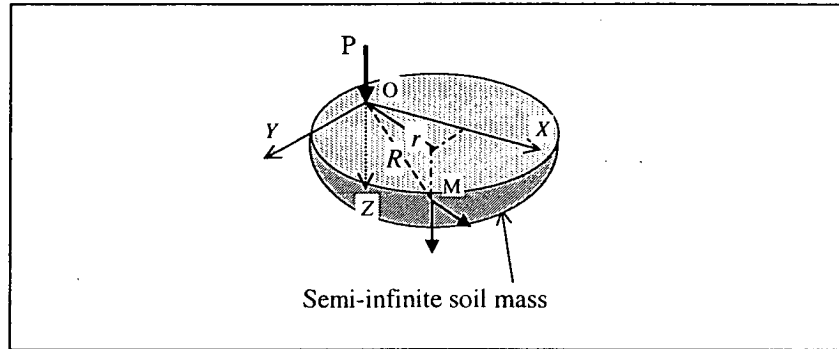


Figure 2.1 Illustration of Boussinesq's approach over a semi-infinite soil mass bearing a normal load.

Equations 2.6 and 2.7 express Boussinesq's solutions, for the spatial displacements of any point in the soil medium due to the application of a vertical force  $P$  at a point  $O$  normal to the medium.

The components of the displacement solution in the radial-horizontal ( $u_r$ )  $x$  and the vertical ( $w_z$ )  $z$  directions of the cylindrical coordinates are respectively described as:

$$u_r = \frac{P(1+\nu_s)}{2\pi E_s R} \left[ \frac{rz}{R^2} - \frac{(1-2\nu_s)r}{R+z} \right] \quad (2.6)$$

$$w_z = \frac{P(1+\nu_s)}{2\pi E_s R} \left[ 2(1-\nu_s) + \frac{z^2}{R^2} \right] \quad (2.7)$$

where  $\nu_s$  and  $E_s$  are the soil's Poisson's ratio and Young's Modulus, respectively;  $R$  is the distance between the point in the medium where the displacement is required ( $M$ ) and that of the load point ( $O$ ), and  $r$  is the projection of  $R$  on the half-space plane (horizontal plane  $x-y$ ).

Figure 2.2 illustrates Cerruti's (1882), see Poulos and Davis (1974), approach. This represents a horizontal point load applied tangentially to the surface of a homogeneous isotropic elastic half-space medium.

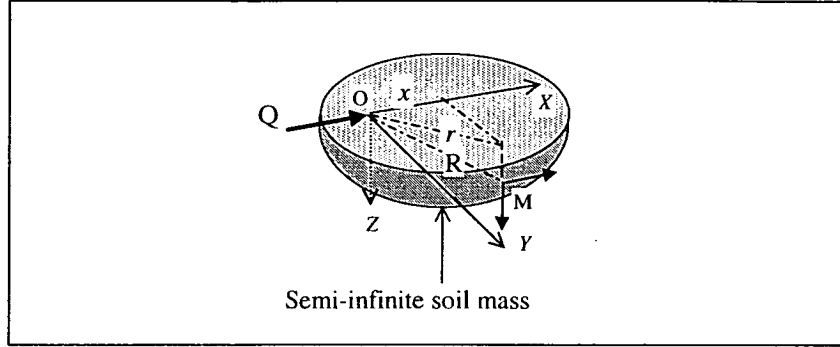


Figure 2.2 Illustration of Cerruti's case over a semi-infinite soil mass bearing a normal load.

Cerruti's solutions for the vertical and horizontal components of the displacement vector of any point ( $M$ ) within the medium due to a single horizontal load ( $Q$ ) applied at point ( $O$ ) tangential to the surface ( $z = 0$ ) of such a medium in Cartesian coordinates system can be expressed as follows:

$$u_r = \frac{Q(1+\nu_s)}{2\pi E_s R} \left[ 1 + \frac{x^2}{R^2} + (1-2\nu_s) \left( \frac{R}{R+z} - \frac{x^2}{(R+z)^2} \right) \right] \quad (2.8)$$

$$w_z = \frac{Q(1+\nu_s)}{2\pi E_s R} \left[ \frac{xz}{R^2} + \frac{(1-2\nu_s)x}{R+z} \right] \quad (2.9)$$

Boussinesq and Cerruti considered applications of point loads on the surface of a homogeneous, isotropic and elastic half-space medium. In practice, the structures actual foundation is located beneath the surface of such a medium and the external forces are applied in the medium; this is known as Mindlin's solutions (1936), which is illustrated in Figure A2.1 in Appendix A2.

Mindlin's solutions are provided in detail in Appendix A2.1.1 (Equations A2.1 to A2.5). These can be used to evaluate the horizontal and vertical displacements at any point ( $F$ ), in the interior of the medium due to both vertical and horizontal forces ( $P$  and  $Q$  respectively). Application of a horizontal moment ( $M_y$ ) to the medium

can be treated as a combination of a couple of forces in the vertical direction. Details of this treatment are explained in the next chapter.

### 2.3.2.1. Further Solutions Associated with Isotropic Elastic Half-space

Where the dimensions of the footings are relatively large, as opposed to a point load, a different approach can be taken.

In applications where the contact area is a rectangle, many authors have analysed the problem as a rectangular plate resting on an isotropic elastic medium and have integrated Boussinesq's Equation 2.7 over the foundation plate area (see Poulos and Davis 1974). If a rectangular rigid foundation of length  $L = 2a$ , width  $B = 2b$  and area  $A$  is considered. The normal point load  $P_z$  is uniformly distributed over the foundation area. Thus

$$P_z = pBL = pA \quad (2.10)$$

The load is applied to an infinitesimal area of  $d\xi \times d\eta$  located in the  $x - y$  plane at distances  $\xi$  and  $\eta$  from the origin of the coordinating system and  $x$  and  $y$  directions respectively such that:

$$dP_z = pd\xi d\eta \quad (2.11)$$

The principle of superposition is applied, and the vertical deflection ( $w$ ) of a general point ( $x, y$ ) under a rectangular footing is obtained as:

$$w(x, y) = \frac{(1 - \nu_s^2)}{\pi E_s} \int_{-a}^a \int_{-b}^b \frac{p(\xi, \eta) d\xi d\eta}{\sqrt{(x - \xi)^2 + (y - \eta)^2}} \quad (2.12)$$

The deflection at the centre point of the above foundation ( $w_c$ ) is given by:

$$w_c = \frac{qB(1 - \nu_s^2)}{\pi E_s} \left( \sinh^{-1} \frac{L}{B} + \frac{L}{B} \sinh^{-1} \frac{B}{L} \right) \quad (2.13)$$

Milovic (1992) described the vertical deflection of any surface point at coordinate  $(x, y, 0)$  as follows

$$w_{(x,y,0)} = \frac{(1-\nu_s^2)p}{E_s\pi} \left[ (b-y) \ln \frac{(a-x)^2 + (b-y)^2 + (a-x)}{(a-x)^2 + (b+y)^2 - (a-x)} + (b+y) \ln \frac{(a+x)^2 + (b-y)^2 + (a-x)}{(a+x)^2 + (b+y)^2 - (a-x)} \right. \\ \left. + (a-x) \ln \frac{(a-x)^2 + (b-y)^2 + (b-y)}{(a-x)^2 + (b+y)^2 - (b+y)} + (a+x) \ln \frac{(a+x)^2 + (b-y)^2 + (b-y)}{(a+x)^2 + (b+y)^2 - (b+y)} \right] \quad (2.14)$$

Equation 2.14 for the vertical surface deflection beneath the centre of a rectangular plate ( $x = y = z = 0$ ) simplifies to:

$$w_c = \frac{2p(1-\nu_s^2)}{\pi E_s} \left[ a \ln \frac{a^2 + b^2 + b}{a^2 + b^2 - b} + b \ln \frac{a^2 + b^2 + a}{a^2 + b^2 - a} \right] \quad (2.15)$$

Gorbunov-Possadov (1949, see Milovic 1992) assumed the contact stress at the interface in the form of Equation 2.16 and proposed that the deflection of the plate can be expressed as a double power series. Contact stress ( $q_p$ ) due to a concentrated load  $P$  could be described as:

$$q_p(x, y) = \sum_{i=0}^n \sum_{j=0}^{n-1} a_{ij} x^i y^j \quad (2.16)$$

where  $a_{ij}$  is a coefficient used to define the contact stress at any point with coordinates  $(x, y)$  of the foundation area. The vertical plate deflection is given as:

$$w(x, y) = \sum_{u=0}^{\infty} \sum_{v=0}^{\infty} A_{uv} x^u y^v \quad (2.17)$$

In the above equation, coefficient  $A_{uv}$  is linearly dependent on the coefficient  $a_{ij}$  used to describe the contact stresses.



Alternatively, for the deflection of any point of foundation of any rigidity, Fraser and Wardle (1976) obtained:

$$w = \frac{pB(1-\nu_s^2)}{E_s} I_w \quad (2.18)$$

where  $B$  is the width of foundation,  $p$  the applied uniform load, and  $I_w$  is a dimensionless coefficient that accounts for the variation of foundation plate stiffness ( $K_F$ ) for rectangular and square foundations. In their solution, with a central point load  $P_0$  acting on a rigid rectangular foundation, the displacement is given by the following equation.

$$w = \frac{P_0}{\sqrt{4ab}} \frac{(1-\nu_s^2)}{E_s} I_w \quad (2.19)$$

where  $I_w$  is provided in tables for different values of  $a$  and  $b$  (Fraser and Wardle 1976).

Applying Boussinesq's solution to consider a distributed load over a circle, the vertical deflection was obtained by:

$$w(r) = \frac{(1-\nu_s^2)}{\pi E_s} \int_0^{a_0} \int_0^{2\pi} \frac{q(\xi) \xi d\xi d\varphi}{\sqrt{r^2 + \xi^2 - 2r\xi \cos\varphi}} \quad (2.20)$$

where  $a_0$  is the radius of the foundation plate,  $\xi$  is the distance of the element in the  $x$  direction from the center,  $\varphi$  and  $d\varphi$  are the angles from the origin and the infinitesimal angle circumscribed by the element respectively.

Egorov (1958) considered the flexibility of foundations implementing elliptic integrals for the calculation of displacements. In that attempt, the following vertical displacement of a flexible circular foundation was obtained.

$$w(r,0) = \frac{2pR(1-\nu_f^2)}{\pi E} [(1-t)K(k) + (1+t)E(k)] \quad (2.21)$$

where  $k = \frac{4t}{(t+1)^2}$ ,  $t = \frac{r}{R}$ ,  $K(k)$  and  $E(k)$  are the complete elliptic integrals of the first and the second kind respectively.

## 2.4 CROSS-ANISOTROPIC ELASTIC HALF-SPACE MODELS

The chronological order of some researchers studying these models is Michell (1900), then Wolf (1935), Quinlan (1949), Koning (1957), Anon (1960), Barden (1963), Lekhnitskii (1963), Urena et al. (1966), Gerrard and Harrison (1970a), and Gerrard and Wardle (1973). Some of these limited their studies solely to the effect of different types of vertical load (such as point load, uniform load, parabolic, inverted parabolic and uniform displacement) on half-space.

A homogeneous anisotropic model in general is associated with a soil medium where Young's modulus and Poisson's ratio vary in different directions. In case of cross-anisotropy, the elastic properties are the same in all directions normal to the axis of symmetry, where this axis is assumed to be vertical, and these properties are different from those in a direction parallel to the axis. This form of anisotropy corresponds to that observed in natural or placed soil and rock deposits formed under the action of predominantly vertical forces.

In a cross-anisotropic medium, the following parameters can be defined:

- $E_h, E_v$  : Young's moduli in the horizontal and vertical directions
- $\nu_h$  : Poisson's ratio for the effect of horizontal stress on complementary horizontal strain
- $\nu_{hv}$  : Poisson's ratio for the effect of horizontal stress on vertical strain
- $\nu_{vh}$  : Poisson's ratio for the effect of vertical stress on horizontal strain
- $F_v$  : Shear modulus for vertical shear stresses

Of these parameters only five are independent, and the following relationship applies

$$\frac{E_h}{E_v} = \frac{\nu_{hv}}{\nu_{vh}} \quad (2.22)$$

Alternatively, the five independent elastic quantities can be expressed by other parameters namely:  $a, b, c, d$  and  $f$ , which are defined in Appendix A2.1.2 (Equations A2.6 to A2.10).

According to Hearmon (1961, see Gerrard and Wardle 1973) and Pickering (1970, see Gerrard and Wardle 1973); the condition that the strain energy should be positive, imposes restrictions on the elastic constants. These restraints are expressed in terms  $a, b, c, d$  and  $f$  as follows:

$$a > 0; d > 0; f > 0; a^2 > b^2; (a + b) d > c^2; ad > c^2 \quad (2.23a)$$

where  $a, b, c, d$  and  $f$  are elastic relations and are defined in Appendix A2.1.2 (Equations A2.6 to A2.10). Such restrictions can also be expressed in terms of the elastic moduli and Poisson's ratios as follows.

$$E_h > 0; E_v > 0; F_v > 0; 1 > \nu_h > -1; 1 - \nu_h - 2\nu_{hv} \nu_{vh} > 0 \quad (2.23b)$$

Gerrard and Harrison (1970a) provided complete solutions for the stress, displacement and strain distributions within the medium due to the application of (pressure) loads over a circular area. The method used to obtain the solutions was based on the application of integral transform techniques to the elasticity problem solved by Sneddon (1951). Further discussion is given in section 2.4.2.1.

Gerrard and Wardle (1973) obtained solutions for the application of point loads (from the general forms) that were obtained for distributed loads over an infinitesimal small circular area (section 2.4.1). The numerical approach was based on the application of integral transform techniques to the elasticity problem solved by Sneddon (1951) and Tranter (1966) (see Gerrard and Wardle 1973). To obtain these solutions, in particular for point loads and to express the stress distribution, the Dirac delta function (Sneddon 1951, see Gerrard and Wardle 1973) was introduced. Some detail on how the Dirac delta function is defined and used to express the stress distribution is available in Appendix A2.1.2.1.

The expressions for displacements were developed in terms of integrals of products of trigonometric functions, Bessel functions (Watson 1944) and exponential

equations (Abramowitz and Stegun 1970). Further discussion is given in section 2.1.2.2.

The form of the solutions depends on the nature of anisotropy as reflected by values of  $\alpha^2$  and  $\beta^2$ , both of which are functions of the elastic properties only. Parameters  $\alpha$  and  $\beta$  are defined as follows:

$$\alpha^2 = \frac{\sqrt{ad} (\sqrt{ad} + c + f)}{2fd} \quad (2.24)$$

where  $\alpha$  is the positive root, and

$$\beta^2 = \frac{ad - c^2 - f(\sqrt{ad} + c)}{2fd} \quad (2.25)$$

$\beta^2$  is not restricted by Equation 2.23a.

The conditions imposed by strain energy in Equation 2.23a are sufficient but not necessary when  $\alpha^2 > 0$ . As a result, in a cross-anisotropic material for each load condition, due to the value of  $\beta^2$ , three possible separate cases can occur. These are as follows:

$$a) \quad \alpha^2 > 0, \quad \beta^2 > 0 \quad (2.26)$$

$$b) \quad \alpha^2 > 0, \quad \beta^2 < 0 \quad (2.27)$$

$$c) \quad \alpha^2 > 0, \quad \beta^2 = 0 \quad (2.28)$$

In a cross-anisotropic medium, when elasticity moduli are such that  $\alpha^2 = 1$  and  $\beta^2 = 0$ , the soil is then simplified to one with isotropic properties. The required coefficients for such a case are provided in Appendix A2.1.2.2.

A cylindrical coordinates system  $(r, \theta, z)$  is introduced for presenting the solutions for displacement in a cross-anisotropic soil. The external load is distributed over a circle, and this system has its origin at the centre of that circle as shown in Figure

2.3. In this system, the  $z$  axis is vertically downwards, the  $r$  axis is radially outwards, and  $\theta$  is the horizontal angle measured in a clockwise direction. The axis of symmetry is assumed to be vertical.

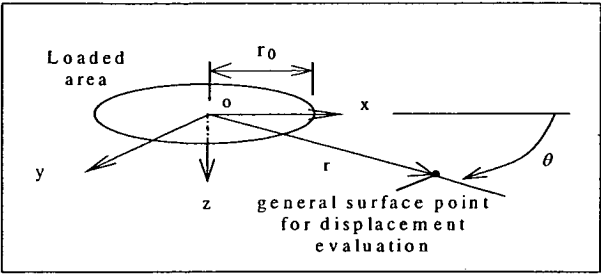


Figure 2.3 Cylindrical coordinates and an infinitesimal circular loaded area (radius  $r_0$ )

In this study, the surface displacements at two points are considered: the point beneath the loaded area ( $z = 0$  and  $r \leq r_0$ ), and that outside the loaded area ( $z = 0$  and  $r > r_0$ ). Parameter  $r$  is the (radial) distance of the point from where the displacement is required to the load center.

### 2.4.1. Points of Load Application and Displacement Distinct

Solutions given for the displacements are a combination of two parts: one part is a constant that corresponds to the integrals due to the geometry of the problem, and the other part is a coefficient that consists of the load. The former appears in constants  $g, h, i, j, s, t$ , and the latter in load coefficients  $L, P$  or  $S$ , which these constants and coefficients are respectively defined in Appendix A2.1.2.2.

The author has employed the solution presented by Gerrard and Wardle (1973) in this thesis (Figure 2.4).

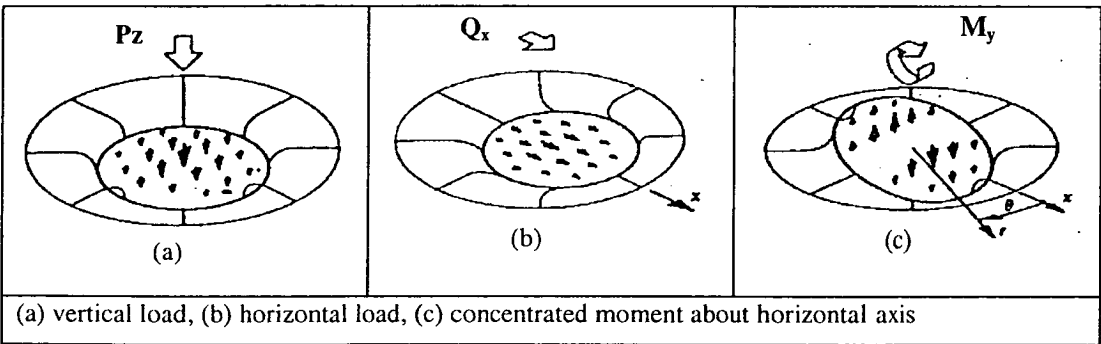


Figure 2.4 Illustration of load cases considered by Gerrard and Wardle (1973)

Three different concentrated load types are considered: two forces and one moment.

These loads are as follows:

- a) Vertical point load  $P_z$
- b) Horizontal point load  $Q_x$
- c) Concentrated moment  $M_y$  about the horizontal axis.

The main equations for displacements are presented below (sections 2.4.1.1 to 2.4.1.3). All the required coefficients are provided in Appendix A2.1.2.2.

### 2.4.1.1. Displacements due to a Vertical Point Load

Solutions for the horizontal ( $u$ ) and the vertical ( $w$ ) displacements due to a vertical point load are as follows:

i. If  $\beta^2 > 0$ , then:

$$u = g_3 L_{020}(\phi z) - g_4 L_{020}(\rho z) \quad (2.29)$$

$$w = g_1 L_{000}(\phi z) - g_2 L_{000}(\rho z) \quad (2.30)$$

ii. When  $\beta^2 < 0$ , then:

$$u = -i_3 {}_c L_{020} - i_4 {}_s L_{020} \quad (2.31)$$

$$w = i_1 {}_c L_{000} + i_2 {}_s L_{000} \quad (2.32)$$

iii. Case of  $\beta^2 = 0$ :

$$u = -s_3 L_{020}(\alpha z) + s_4 z L_{022}(\alpha z) \quad (2.33)$$

$$w = s_1 L_{000}(\alpha z) + s_2 z L_{002}(\alpha z) \quad (2.34)$$

Constants  $g_1, \dots, g_4, i_1, \dots, i_4, s_1, s_3$  and load coefficients  $L_{000}(\phi z), L_{020}(\phi z), {}_c L_{000}, {}_s L_{000}, {}_c L_{020}, {}_s L_{020}$  are given in Appendix A2.1.2.2.

### 2.4.1.2. Displacements due to a Horizontal Point Load

Solutions for the horizontal ( $u$ ) and the vertical ( $w$ ) displacements due to a horizontal point load are given:

i. If  $\beta^2 > 0$ , then:

$$u = \cos \theta \{h_3 S_{020}(\phi z) - h_4 S_{020}(\rho z) + h_{10} P_{020}(\gamma z)\} \quad (2.35)$$

$$w = \cos \theta \{h_1 L_{020}(\phi z) - h_2 L_{020}(\rho z)\} \quad (2.36)$$

ii. When  $\beta^2 < 0$ , then:

$$u = \cos \theta \{j_3 {}_c S_{020} + j_4 {}_s S_{020} + j_{11} P_{020}(\gamma z)\} \quad (2.37)$$

$$w = \cos \theta \{j_1 {}_c L_{020} + j_2 {}_s L_{020}\} \quad (2.38)$$

iii. For  $\beta^2 = 0$ :

$$u = \cos \theta \{-t_3 S_{020}(\alpha z) + t_4 z S_{022}(\alpha z) + t_{10} P_{020}(\gamma z)\} \quad (2.39)$$

$$w = \cos \theta \{t_1 L_{020}(\alpha z) + t_2 z L_{022}(\alpha z)\} \quad (2.40)$$

Constants  $j_1, \dots, j_4, j_{11}, h_1, \dots, h_4, h_{10}, t_1, t_3, t_{10}$  and load coefficients  $S_{020}(\alpha z), P_{020}(\gamma z), L_{022}(\gamma z), S_{020}, {}_c S_{020}, {}_s S_{020}$  are provided in Appendix A2.1.2.2.

### 2.4.1.3. Displacements due to a Moment Load about Horizontal Axis $z$

Solutions for the horizontal ( $u$ ) and the vertical ( $w$ ) displacements due to a concentrated moment about the horizontal axis ( $y$ ) are as follows:

i. If  $\beta^2 > 0$ , then:

$$u = \cos \theta \{g_3 S_{120}(\phi z) - g_4 S_{120}(\rho z)\} \quad (2.41)$$

$$w = \cos \theta \{g_1 L_{120}(\phi z) - g_2 L_{120}(\rho z)\} \quad (2.42)$$

ii. When  $\beta^2 < 0$ , then:

$$u = \cos \theta (i_3 {}_c S_{120} - i_4 {}_s S_{120}) \quad (2.43)$$

$$w = \cos \theta (i_1 {}_c L_{120} + i_2 {}_s L_{120}) \quad (2.44)$$

iii. For  $\beta^2 = 0$ :

$$u = \cos \theta \{-s_3 S_{120}(\alpha z) + s_4 z S_{122}(\alpha z)\} \quad (2.45)$$

$$w = \cos \theta \{s_1 L_{120}(\alpha z) + s_2 z L_{122}(\alpha z)\} \quad (2.46)$$

Constants  $I_{120}(\alpha z)$ ,  ${}_s I_{120}$  and load coefficients  ${}_c L_{120}$ ,  $S_{120}(\alpha z)$ ,  ${}_c S_{120}$ ,  ${}_s S_{120}$  are provided in Appendix A2.1.2.2.

## 2.4.2. Points of Load Application and Displacement Coincident

In this case the points of load application and displacement evaluation are coincident, and the applied load is idealised as a distributed load acting in the vicinity of that application point. For this case, Gerrard and Harrison (1970a) solutions can be applied. In these solutions, sign convention is as presented earlier in Figure 2.3, and necessary conversion expressions are introduced to evaluate the concentrated equivalent of the applied distributed loads. These terms are presented in the next chapter. The three following load cases are considered (Figure 2.5):

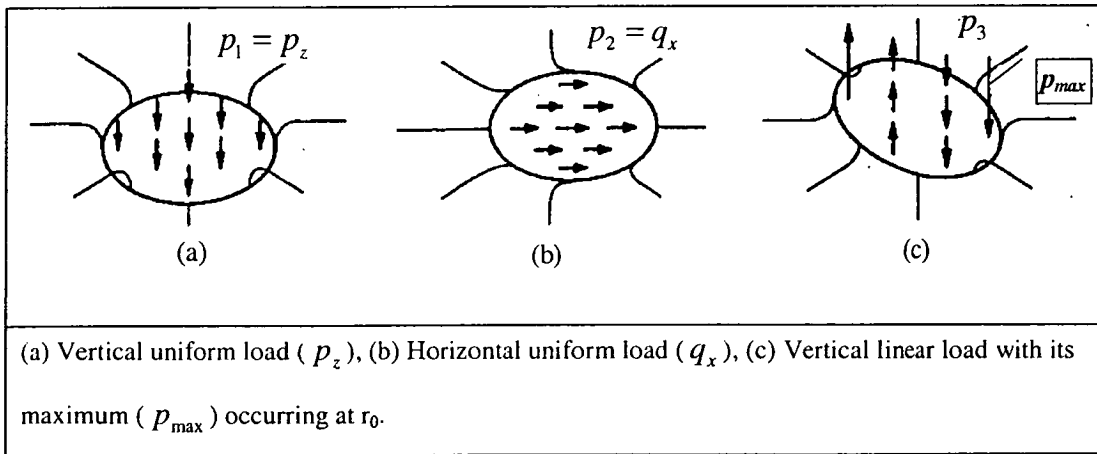


Figure 2.5 Stress resolution for the load cases considered in this study (Gerrard and Harrison 1970a).



- a) Vertical uniform load ( $p_1$ );
- b) Horizontal uniform unidirectional shear load ( $p_2$ );
- c) Vertical linear load ( $p_3$ ) with its maximum,  $p_{\max}$ , occurring at  $r_0$ .

The main equations for displacements are presented below (sections 2.4.2.1 to 2.4.2.3). All the required coefficients and parameters are provided in Appendix A2.1.2.2.

### 2.4.2.1. Displacements Due to a Vertical Uniform Load

Expressions for the horizontal and vertical displacements due to a vertical uniform load are as follows:

i.  $\beta^2 > 0$ :

$$u = p_1 r_0 [g_3 I_{200}(\phi z) - g_4 I_{200}(\phi z)] \quad (2.47)$$

$$w = p_1 r_0 [g_1 I_{200}(\phi z) - g_2 I_{200}(\phi z)] \quad (2.48)$$

ii.  $\beta^2 < 0$ :

$$u = p_1 r_0 [-i_3 {}_c I_{200} - i_4 {}_s I_{220}] \quad (2.49)$$

$$w = p_1 r_0 [i_1 {}_c I_{200} + i_2 {}_s I_{200}] \quad (2.50)$$

iii.  $\beta^2 = 0$ :

$$w = p_1 r_0 [-s_3 I_{220}(\alpha z) + s_4 z I_{222}(\alpha z)] \quad (2.51)$$

$$w = p_1 r_0 [s_1 I_{200}(\alpha z) + s_2 z I_{202}(\alpha z)] \quad (2.52)$$

Coefficients  $g_1, \dots, g_4$ ,  $i_1, \dots, i_4$ ,  $s_1, s_3$ ,  $I_{200}(\phi z)$ ,  $I_{220}(\phi z)$ ,  ${}_c I_{200}$ ,  ${}_s I_{200}$ ,  ${}_c I_{220}$ ,  ${}_s I_{220}$ , and  $I_{222}(\alpha z)$  are provided in Appendix A2.1.2.

### 2.4.2.2. Displacements Due to a Horizontal Uniform Load

Solutions for the horizontal and the vertical displacements due to a uniform horizontal shear load are as follows:

i.  $\beta^2 > 0$ :

$$u = p_2 \cos \theta r_0 [h_3 \cdot M_{220}(\phi z) - h_4 \cdot M_{220}(\rho z) + h_{10} \cdot A_{220}(\gamma z)] \quad (2.53)$$

$$w = p_2 \cos \theta r_0 [h_1 I_{220}(\phi z) - h_2 I_{220}(\rho z)] \quad (2.54)$$

ii.  $\beta^2 < 0$ :

$$u = p_2 \cos \theta r_0 [j_3 \cdot M_{220} + j_4 \cdot M_{220} + j_{11} A_{220}(\gamma z)] \quad (2.55)$$

$$w = p_2 \cos \theta r_0 [j_1 \cdot I_{220} + j_2 \cdot I_{220}] \quad (2.56)$$

iii.  $\beta^2 = 0$ :

$$u = p_2 \cos \theta r_0 [-t_3 M_{220}(\alpha z) + t_4 z M_{222}(\alpha z) + t_{10} A_{220}(\gamma z)] \quad (2.57)$$

$$w = p_2 \cos \theta r_0 [t_1 I_{220}(\alpha z) + t_2 z I_{222}(\alpha z)] \quad (2.58)$$

Coefficients  $h_1, \dots, h_4, j_1, \dots, j_4, t_1, t_3, t_{10}, A_{220}(\gamma z), I_{220}(\alpha z), {}_s I_{220}, {}_c I_{220}, M_{220}(\alpha z), {}_c M_{220}, {}_s M_{220}, M_{222}(\gamma z)$  are given in Appendix A2.1.2.2.

### 2.4.2.3. Displacements Due to a Vertical Linear Load

Expressions of solutions for the horizontal and the vertical displacements due to a linear vertical load (resolution of application of moment  $M_y$ ) are as follows:

i.  $\beta^2 > 0$ :

$$u = p_3 \cos \theta r_0 [g_3 \cdot M_{420}(\phi z) - g_4 \cdot M_{420}(\rho z)] \quad (2.59)$$

$$w = p_3 \cos \theta r_0 [g_1 I_{420}(\phi z) - g_2 I_{420}(\rho z)] \quad (2.60)$$

ii.  $\beta^2 < 0$ :

$$u = p_3 \cos \theta r_0 [-i_3 {}_c M_{420} - i_4 {}_s M_{420}] \quad (2.61)$$

$$w = p_3 \cos \theta r_0 [i_1 {}_c I_{420} + i_2 {}_s I_{420}] \quad (2.62)$$

iii.  $\beta^2 = 0$ :

$$u = p_3 \cos \theta r_0 [-s_3 M_{420}(\alpha z) + s_4 z M_{422}(\alpha z)] \quad (2.63)$$

$$w = p_3 \cos \theta r_0 [s_1 I_{420}(\alpha z) + s_2 z I_{422}(\alpha z)] \quad (2.64)$$

Coefficients  $g_1, \dots, g_4, i_1, \dots, i_4, s_1, s_3, I_{420}(\alpha z), {}_c I_{420}, {}_s I_{420}, I_{422}(\alpha z), {}_c M_{420}, {}_s M_{420}, M_{422}(\alpha z), M_{420}(\alpha z)$  are defined in Appendix A2.1.2.2.

## 2.5. NON-HOMOGENEOUS ISOTROPIC ELASTIC HALF-SPACE MODEL

Holl (1940) developed a general form of Boussinesq's (1885) classical solutions for the problem of a vertical point load ( $P_z$ ) and a horizontal point load ( $Q_x$ ) acting on the surface of a non-homogeneous half-space.

In this soil idealisation, modulus of elasticity varies with depth such that:

$$E = E_0 z^\lambda \quad (2.65)$$

where  $E_0$  is the modulus of elasticity at unit depth and

$$\lambda = n - 3 = \frac{1}{\nu} - 2 \quad (2.66)$$

In Equations 2.65 and 2.66, for  $n = 3$ ,  $\lambda = 0$ ,  $\nu = 0.5$ , the case reduces to Boussinesq's solution and when  $n = 4$ ,  $\lambda = 1$ ,  $\nu = 1/3$ , it corresponds to Gibson's solution (1967).

In his research, Holl developed parametric solutions for radial, tangential and shear stresses in cylindrical coordinate systems in which for some specific parameters, those solutions were simplified to Gibson's model.

For a non-homogeneous elastic half-space medium with a linearly varying shear model illustrated in Figure 2.6, Gibson (1967) suggested that the shear modulus ( $G_z$ ) take the form:

$$G(z) = G_0 + m z \quad (2.67)$$

where  $G_0$  and  $m$  are the soil shear modulus at the surface of elastic half-space and a constant that describes the linear increase of the soil property with depth respectively.

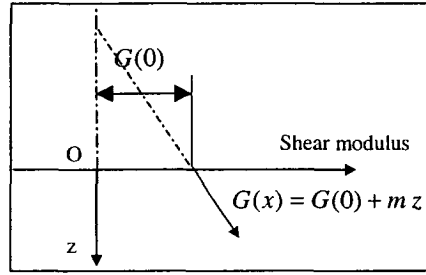


Figure 2.6 Variation of shear modulus in an inhomogeneous soil idealised by Gibson (1967).

The displacements due to a vertical uniform surface load ( $p$ ), over a circular area ( $r \leq b$ ), for a constant Poisson's ratio ( $\nu = 1/3$ ), are expressed by Awojobi and Gibson (1973). In their method new dependant variables were applied:

$$\Omega(x, z) = \frac{1}{h} \left( \frac{\partial u}{\partial z} - \frac{\partial w}{\partial x} \right) \quad (2.68)$$

and

$$\epsilon(x, z) = \frac{1}{h} \left( \frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} \right) \quad (2.69)$$

where  $\Omega$  and  $\epsilon$  are odd and even functions and are expressed in terms of the following Fourier transforms:

$$\Omega(x, z) = \frac{2}{\pi} \int_0^{\infty} \Omega^*(\xi, z) \sin x \xi d\xi \quad (2.70)$$

$$\epsilon(x, z) = \frac{2}{\pi} \int_0^{\infty} \epsilon^*(\xi, z) \cos x \xi d\xi \quad (2.71)$$

This approach leads to the general solutions for horizontal and vertical displacements as follows:

$$u(x, z) = \frac{2}{\pi} \int_0^{\infty} u^*(\xi, z) \sin x \xi d\xi \quad (2.72)$$

$$w(x, z) = \frac{2}{\pi} \int_0^{\infty} w^*(\xi, z) \cos x \xi d\xi \quad (2.73)$$

Specifically, when there is an axial symmetry applicable to the problem, Fourier kernel  $\frac{2}{\pi} \frac{\sin b \xi \cos x \xi}{\xi}$  is replaced by the Hankel kernel  $b J_0(r \xi) J_1(b \xi)$ . The displacement on the half-space is obtained

$$w(r, 0) = \frac{p b}{2 G_0} \int_0^{\infty} \frac{J_0(r \xi) J_1(b \xi)}{\xi} f(\beta \xi) d\xi \quad (2.74)$$

where

$$f(\beta \xi) = f(t_0) = \frac{t_0 [K_1^2(t_0) - K_0^2(t_0)] + K_0(t_0) K_1(t_0)}{t_0^2 [K_1^2(t_0) - K_0^2(t_0)] + 2 K_1^2(t_0) - t K_0(t_0) K_1(t_0)} \quad (2.75)$$

where  $K_0$ , and  $K_1$  are modified Bessel functions;  $t_0 = \xi \beta$ ,  $\beta = G_0 / m$ .  $J_0$  and  $J_1$  are the Bessel functions of the first kind of order zero and unity. The parameter  $r$  and  $b$  are the distance from the origin of the coordinate system, and the radius of the loaded area respectively.

Dempsey and Li (1995) applied the above solution for a vertical point load ( $P$ ) at the origin of the coordinate system, where  $b$  tends to zero,  $p \pi b^2$  tends to  $P$ , and Equation 2.74 simplifies to:

$$w_0(r) = \frac{P}{4\pi G_0} \int_0^\infty J_0(r\xi) f(\beta\xi) d\xi \quad (2.76)$$

In the analytical evaluation of the above integral, the following solution is obtained

$$w_0(r) = \frac{P}{4\pi G_0} \left[ \frac{4}{3r} - \frac{5}{3\beta} \ln \frac{\beta + \sqrt{\beta^2 + r^2}}{r} + \sum_{m=1}^M \frac{\alpha_m}{\sqrt{(m\gamma\beta)^2 + r^2}} \right] \quad (2.77)$$

where  $\alpha_m$  and  $\gamma$  are coefficients that are determined by the least-squares error criterion to minimise the error and  $M$  is the number of terms to be considered.

Further details of the solutions are beyond the scope of this study.

## 2.6. METHODS OF OBTAINING SOIL PARAMETERS

Soil moduli of elasticity vary over a wide range while Poisson's ratio variation is much more limited. In an isotropic linear elastic soil it is possible to determine the elastic constants  $E$  (Young's modulus) and  $\nu$  (Poisson's ratio) from a single uniaxial test. These constants are then used to calculate the relationship between stress and strain for other types of tests. In uniaxial tests, the concept of theory of elasticity is considered. If an axial load  $\sigma_z$  is applied to an elastic cylinder, there will be a vertical compression and a lateral expansion where the modulus of elasticity is obtained from the stress-strain curve.

Application of shear stress  $\tau_{zx}$  to an elastic cube causes a shear distortion that can be used to obtain the shear modulus. Details of these tests are given in Appendix A2.2.1. In situ tests of Standard Penetration Test (SPT) and Cone Penetration Test (CPT) tend to use empirical correlations to obtain  $E_s$ . Other in situ tests such as the pressuremeter, the flat dilatometer, the Iowa stepped blade tend to obtain more direct measurements of  $E_s$ . The value of stress-strain modulus  $E_s$  obtained from these tests is generally the horizontal value, however the vertical value is usually needed for settlements. Most soils are anisotropic, so the horizontal  $E_{sh}$  value may be

considerably different from the vertical value  $E_{shv}$ . Overconsolidation may also alter the vertical and horizontal values of strain modulus.

Because the laboratory values of  $E_s$  are expensive to obtain and are generally not very good anyway owing to sampling disturbance, the SPT and CPT have been widely used to obtain the stress-strain modulus  $E_s$  resulting from empirical equations and/or correlations. Tables 2.1 and 2.2 (Bowles 1997) give a number of equations for possible use in several test methods.

Soil	SPT (Standard Penetration Test)	CPT (Cone Penetration Test)
Sand (normally consolidated)	$E_s = 500 (N + 15)$ $= 7000 \sqrt{N}$ $= 6000 N$ $E_s = (15\ 000 \text{ to } 22\ 000) \cdot \ln (N)$	$E_s = (2 \text{ to } 4) q_u$ $= 8000 \sqrt{q_c}$ $E_s = 1.2 (3D_r^2 + 2) q_c$ $E_s = (1 + D_r^2) q_c$
Sand (saturated)	$E_s = 250 (N + 15)$	$E_s = F q_c$ $e = 1.0 \quad F = 3.5$ $e = 0.6 \quad F = 7.0$
Sands, all (normal consolidated)	$E_s = (2600 \text{ to } 2900) N$	
Sand (overconsolidated)	$E_s = 40\ 000 + 1050 N$ $E_{s(OCR)} \approx E_{s,nc} \sqrt{OCR}$	$E = (6 \text{ to } 30) q_c$
Gravelly sand	$E_s = 1200 (N + 6)$ $= 600 (N + 6) \quad N \leq 15$ $= 600 (N + 6) + 2000 \quad N > 15$	
Clayey sand	$E_s = 320 (N + 15)$	$E_s = (3 + 6) q_c$
Silts, sandy silt, or clayey silt	$E_s = 300(N + 6)$	$E_s = (1 \text{ to } 2) q_c$
Soft clay or clayey silt		$E_s = (3 \text{ to } 8) q_c$
$E_s$ in kPa for SPT and units of $q_c$ (cone bearing pressure) for CPT; divide kPa by 50 to obtain ksf. The $N$ (SPT blow count) values should be estimated as $N_{55}$ and not $N_{70}$ (SPT blow count at 55 & 70 percent efficiency). See Bowles, 1997 for definitions of variables.		

Table 2.1 Equations defining stress-strain modulus  $E_s$  by several test methods

SOIL		
Clay and silt	$I_p > 30$ or organic	$E_s = (100 \text{ to } 500)s_u$
Silty and sandy clay	$I_p > 30$ or stiff	$E_s = (100 \text{ to } 1500)s_u$ Again, $E_{s,OCR} \approx E_{s_{nc}} \sqrt{OCR}$ Use smaller $s_u$ – coefficient for highly plastic clay.
Of general application in clay is $E_s = Ks_u$ (units of $s_u$ )		(a)
where $K$ is defined as $K = 4200 - 142.54I_p + 1.73I_p^2 - 0.0071P_p^3$		(b)
and $I_p$ = plasticity index in <b>percent</b> . Use $20\% \leq I_p \leq 100\%$ and round $K$ to the nearest multiple of 10. Another equation of general application is  $E_s = 9400 - 8900I_p + 11\,600I_c - 8800S$ (kPa) (c)  where $I_p$ plasticity index, $I_c$ relative consistency which is defined as $I_c = \frac{w_L - w_N}{I_p}$ ( $w_L$ is liquid limit and $w_N$ is water content), and $S$ is degree of saturation which is defined as the ratio of the volume of water to the total volume of soil voids, expressed as a percentage $S = \frac{V_w}{V_v} \times 100$ (%)		
Use the undrained shear strength $s_u$ in units of $s_u$		

Table 2.2 The commonly used multiplier  $\sqrt{OCR}$  (overconsolidation ratio) used to increase the normally consolidated value of stress-strain modulus  $E_{s,nc}$

Some values and value ranges for Poisson’s ratio and modulus of elasticity in isotropic soil medium are presented in Tables 2.3, 2.4 and 2.5 (Bowles 1997).

Type of soil	$\mu$
Clay, saturated	0.4 - 0.5
Clay, unsaturated	0.1 - 0.3
Sandy clay	0.2 - 0.3
Silt	0.3 - 0.35
Sand, gravelly sand commonly used	0.1 - 1.00 0.3 - 0.4
Rock	0.1 - 0.4 (depend somewhat on type of rock)
Loess	0.1 - 0.3
Ice	0.36
Concrete	0.15
Steel	0.33

Table 2.3 Values and some value ranges for Poisson’s ratio  $\mu$  for selected materials



Soil type	$\mu$
Most clay soils	0.4 - 0.5
Saturated clay soils	0.45 - 0.50
Cohesionless — medium and dense	0.3 - 0.4
Cohesionless — loose to medium	0.2 - 0.35

Table 2.4 The most commonly used values for soils

Soil	$E_s^*$ , MPa
Clay	
Very soft	2 - 15
Soft	5 - 25
Medium	15 - 50
Hard	50 - 100
Sandy	25 - 250
Glacial till	
Loose	10 - 150
Dense	150 - 720
Very dense	500 - 1440
Loess	15 - 60
Sand	
Silty	5 - 20
Loose	10 - 25
Dense	50 - 81
Sand and Gravel	
Loose	50 - 150
Dense	100 - 200
Shale	150 - 5000
Silt	2 - 20
<ul style="list-style-type: none"> <li>* Value range is too large to use an "average" value for design</li> <li>Field values depend on stress history, water content, density and age of deposit</li> </ul>	

Table 2.5 A range of  $E_s$  values (possibly obtained) for the static stress-strain modulus  $E_s$  for selected soils

Other tests (Isotropic compression, Confined compression or Oedometer and Triaxial compression) can be used to obtain soil elastic parameters as described in standard soil mechanics text references (e.g. Lambe and Whitman 1979, Taylor 1948).

Many of the existing studies for assessing the elastic parameters in cross-anisotropic soil are conducted to obtain parameters for road pavement designs. Anisotropy, stress history, natural cementation, and overconsolidation are likely to be very significant

factors in determining  $E_s$ , especially for cohesionless soils. In cohesionless soils cementation is particularly significant; for individual soil grains the effect can be very small, but the statistical accumulation for the mass can have a large effect. Cementation (also called "aging") can be easily lost in recovered cohesionless samples. Drilling disturbances in cohesionless soils for the purpose of performing pressuremeter, dilatometer, or other tests may sufficiently destroy the cementation/aging in the vicinity of the hole to reduce  $E_s$  to little more than an estimate.

Since a specific design strength is required the tests are proposed accordingly. A limited amount of research is available in relation to structures founded on this type of half-space. However, there are a number of field and laboratory tests that can be used indirectly to obtain the soil parameters. Among these laboratory tests, *CBR* or California Bearing Ratio (ASTM 1987 and AS1289) method are utilised to determine the bearing ratio of soil when it is compacted and tested in the laboratory by comparing the penetration load of the soil to a standard material. This method covers the evaluation of the relative strength of the soil (Pavement Design 1992). The *CBR* test can be used to determine the modulus of subgrade reaction. As a resilient technique, the vertical modulus of subgrade can be obtained either from the laboratory testing of conditioned specimens (Thompson and Quentin 1976) or from the following empirical relationship (AUSTROAD, Pavement Design 1992)

$$\text{Vertical Modulus of Elasticity (MPa)} = 10 \text{ CBR} \quad (2.78)$$

This equation is at best an approximation. The modulus is found to vary in a range 5 to 20 *CBR* (Sparks and Potter 1982).

The triaxial test is an alternative laboratory test that can be used. Barret and Smith (1976) found that the modulus of elasticity for a crushed dolerite with clay binder, from repeated load triaxial tests could be adequately predicted by:

$$E = k (\sigma_1 + 2\sigma_3)^m \quad \text{MPa} \quad (2.79)$$

where  $k$  and  $m$  are parameters that depend on the number of load repetitions and they vary in the ranges 75-110 and 0.32-0.50 respectively (Graph A2.3 in Appendix A2),  $\sigma_1$  is the applied compressive stress,  $\sigma_3$  is the orthogonal restraining stress. Further details regarding the above methods can be obtained from Appendix A2.2.

## **2.7 CONCLUDING REMARKS TO CHAPTER**

Several soil models and the corresponding expressions for the force-displacement relation were presented. These expressions were shown in different forms, such as exact solutions (e.g. closed forms) or mathematical series. Relevant soil methods from the above mentioned are implemented in the next chapter to obtain the coefficients of the flexibility matrix for individual cases that are utilised in this thesis.

# CHAPTER THREE

*METHOD OF SOLUTION*

*Elastostatic Interaction Analysis of Frames  
Resting on Homogeneous Elastic Half-space*

# **CHAPTER 3**

## **METHOD OF SOLUTION**

### **3.1. INTRODUCTION**

This chapter elaborates on the methods used to analyse the system of a superstructure and its supporting medium.

Most of the currently designed structures are statically indeterminate. Indeterminate structures are preferable to determinate structures because of the numerous advantages they have during their service life. This indeterminacy may arise as a result of added supports, members, or from the geometry of the structure. For instance, most reinforced concrete buildings are statically indeterminate. The reason for this is that the columns and beams are poured as continuous members through the joints and over supports (monolithic structures). The analysis of statically indeterminate structures generally requires the solution of linear simultaneous equations, the number of which depends on the method of analysis.

Difficult cases of variable soil deposits, loading conditions and complex structures usually necessitate performing a deformation analysis by means of numerical techniques such as the finite element method or the boundary element method. However, some numerical techniques require extensive calculations which can be very time-consuming and therefore they may not be suitable for conducting preliminary design studies.

### **3.2. STRUCTURE-SOIL SYSTEM**

The combination of a structure and its supporting soil medium constitutes a system. The present study is concerned with the interaction that occurs between these components and how they respond to the application of external loads.

The approach taken in the study is as follows.

1. A single plane frame in a vertical plane  $x - y$ , regarded as the main structure, is analysed, and the reaction of soil to the structure in response to the external loads is taken into account (Figure 3.1).
2. A number of such plane frames parallel to the single frame and located on both sides of it are then considered. These frames, which are exposed to external loads similar to loads on the single frame, are related to the single frame only through their interaction via the supporting soil. Support interaction between adjacent frames and the effect of that interaction is analysed using a numerical approach.

The direct stiffness method (DSM) is a displacement-based technique which due to its simplicity, is used in this study. This method can be applied to all types of structures subjected to external loads or to prescribed displacements. DSM is a relatively modern computer oriented method of analysis that is employed for linear elastic analysis of frame structures.

The stiffness equations are derived from the equilibrium of nodes, and the nodal displacements are determined from them. These stiffness equations can be expressed as:

$$[K_{st}]\{\Delta\} = \{P\} \quad (3.1)$$

where  $[K_{st}]$  is the structure stiffness matrix which is square,  $\{\Delta\}$  is the vector of nodal displacements, and  $\{P\}$  represents the vector of external nodal forces. Applying Equation 3.1 to the superstructure with node loads as well as element loads yields:

$$[K_{st}]\{\Delta\} = \{\bar{P}\} - \{P\}^0 \quad (3.2)$$

where  $\{P\}^0$  is the vector composed of fixed end forces and:

$$\{\bar{P}\} = \{P\} + \{R_s\} + \{R_{rs}\} \quad (3.3)$$

where  $\{P\}$  is the vector of nodal loads,  $\{R_s\}$  is the vector of reaction forces and  $\{R_{rs}\}$  is the vector of nodal elastic restraint forces.

The vector  $\{R_s\}$  can be expressed as:

$$\{R_s\} = [K_s]\{\Delta_s\} \quad (3.4)$$

where  $\{R_s\}$  and  $\{\Delta_s\}$  are the vector of interaction forces between the soil and structure and the vector of nodal displacements respectively, and where  $[K_s]$  is the soil stiffness matrix.

Of course, the interaction forces acting on the structure and the soil are equal in magnitude but opposite in direction.

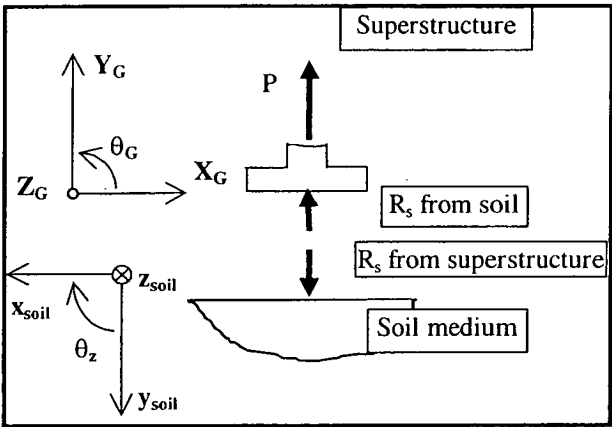


Figure 3.1 Coordinate systems, superstructure, soil medium and the interaction.

Thus, if coordinates on soil are chosen in opposite directions to those of the structure global coordinates, then it can be stated that:

$$\{R_s\} = [K_s]\{\Delta_s\} = -[K_s]\{\Delta\} \quad (3.5)$$

By substituting Equation 3.5 in Equation 3.2, the static system stiffness relationship that takes account of the stiffness of the superstructure and that of the foundation bed is obtained as:

$$[K_{sys}]\{\Delta\} = \{P\} - \{P\}^0 \quad (3.6)$$

where

$$[K_{sys}] = [K_{st}] + [K_s] \quad (3.7)$$

The soil stiffness matrix  $[K_s]$  in this work is obtained as an inverse of the soil flexibility matrix.

The soil flexibility matrix is prepared for the following options:

1. The soil stiffness matrix is associated with the support nodes of a single plane frame structure that does not consider the effect of neighbouring frames or the adjacent support nodes in the same frame. In this case, only the effect of support reactions under individual nodes of a single frame (stand-alone) on the soil is evaluated in the structural analysis (isolated supports).
2. The soil stiffness matrix is prepared to contribute the effect of adjacent support nodes of a single plane frame structure. This incorporates the effects of the neighbouring support nodes of the same frame. The option is available for the study of the individual support interactions of a single frame, and also of the interaction of the other support nodes from the same frame on each other through the soil medium. Support interaction is allowed through the continuous soil medium supporting the structure.
3. The soil stiffness matrix is built to consider the interaction amongst all the support nodes (in the soil). The scope of this interaction is from supports of the same frame and from other frames, which are assumed to be in parallel and to be related to the frame concerned through the soil medium.

These three cases are illustrated in chapter 5.

### **3.3. DIRECT STIFFNESS MATRIX METHOD**

When analysing a structure subjected to static loads, there are two main groups of unknowns. They are the internal force distribution within the structure and the nodal displacements that the structure undergoes.



The Direct Stiffness Method involves solving simultaneous linear equations that describe the force-displacement relations expressed by Equation 3.1. Since the method is well known (Ghali and Neville 1989, Hsieh and Mau 1995) only a brief description is given here. This method can be outlined as follows:

1. The structure is idealised by means of elements connected at nodes. The stiffness equation is developed for these elements depending on the support conditions. The conditions are either zero or non-zero degrees of freedom. To obtain the final structure stiffness matrix, the developed equations are combined and rearranged according to these support conditions.
2. Nodal forces are calculated by considering the fixed end forces due to the member loads and the nodal prescribed displacements. The external node loads are then added to the nodal force vector.
3. Spring stiffnesses are added to the stiffness matrix of the superstructure. Once it has accounted for the contribution of the linear spring supports, the superstructure stiffness matrix is prepared.
4. The soil flexibility matrix is obtained using the force-displacement relations derived for the appropriate soil model. The inverse of this matrix gives the soil stiffness matrix, which is then superimposed on the structure stiffness matrix to obtain the stiffness matrix of the whole system.
5. The stiffness equations of the whole system are solved for the nodal displacements in global coordinates.
6. The internal end forces are then calculated by substituting the nodal displacements into the element stiffness relations.

Based on the above steps, the author has developed a computer analysis package *SASIAP* (Soil And Structure Interaction Analysis Package) using Turbo Pascal language.

### **3.4. SUPERSTRUCTURE CONTRIBUTION IN ANALYSIS**

The superstructure is the entire construction that stands above the soil medium. The above steps of the Direct Stiffness Matrix are applied to the superstructure in order to analyse the whole system.

### 3.4.1. Assumptions in Superstructure Analysis

In the analysis of the superstructure, Hooke's law governs the conditions. Hence, only linear elastic parameters are considered for displacements, stresses and strains. The displacements are small and do not significantly affect the geometry of the superstructure. Members are prismatic and symmetric with respect to the plane of the structure. As the structure is located in a plane, three degrees of freedom are defined for individual nodes (nodal displacements). The shear deformation effects on the elements are ignored and cross-sections remain plane during bending.

### 3.5. SOIL CONTRIBUTION IN ANALYSIS

Figure 3.2 shows the corresponding interaction forces acting on the footing nodes and the soil medium.

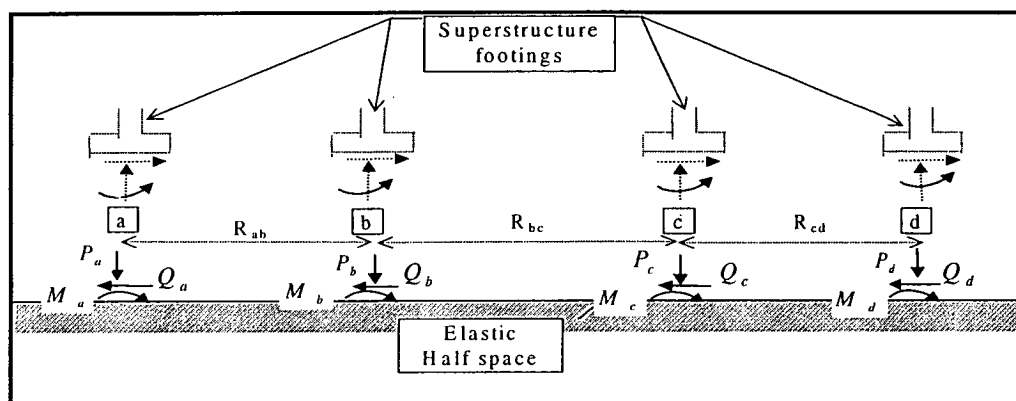


Figure 3.2 Illustration of soil and superstructure nodal reactions at the interface.

The reactions at the footing supports are considered as internal forces, which are equal and in opposite directions. The sign conventions shown in Figures 3.1 and 3.2 are adopted for developing the soil flexibility coefficients.

As described in Chapter 2, the force-displacement relations are used at the soil-structure interface to develop the soil stiffness matrix. In the following sections, the effects of horizontal and vertical forces, and moments about horizontal axis applied to the soil are taken into consideration. In this project, two types of soil consisting of homogeneous elastic half-space with isotropic and cross-anisotropic behaviour are elaborated.

### 3.5.1. Assumptions in Soil Consideration

The supporting medium is modelled as a homogeneous elastic half-space with either isotropic or cross-anisotropic properties, and the vertical axis is the axis of symmetry for the elastic properties in a horizontal direction in a cross-anisotropic case. The soil remains in contact with the superstructure and the nodes at support positions in the structure coincide with those of the elastic half-space.

### 3.5.2 Development of Soil Stiffness Matrix

The soil stiffness matrix is obtained by inverting the soil flexibility matrix. The flexibility equation is defined in a similar way to the stiffness equation, with force and displacement interchanged throughout, and refers to the displacement at some point on the surface of the medium caused by a unit force at the same point or at some other point. Flexibility equations obtained from the assembling of such relations are then used to develop the soil stiffness matrix.

The flexibility relationships depend upon the soil properties and the distance between two points in the soil. These points are where the displacement takes place and where the force acts, and the relationship of these points to each other determines the calculation. Based on whether this distance tends to zero (displacement concerned under the load) or not (displacement concerned at some other point), two categories of flexibility coefficients are identified as group *B* and group *A* respectively and are shown in Figure 3.3.

$$[F]_{ab} = \begin{bmatrix} [F_{aa}] & [F_{ab}] \\ [F_{ba}] & [F_{bb}] \end{bmatrix} = \left[ \begin{array}{c|ccc} & & A_1 & A_2 & A_3 \\ & & A_4 & A_5 & A_6 \\ & & A_7 & A_8 & A_9 \\ \hline & & B_1 & B_2 & B_3 \\ & & B_4 & B_5 & B_6 \\ & & B_7 & B_8 & B_9 \end{array} \right]$$

Figure 3.3 Illustration of different sub-matrix groups in soil flexibility matrix between two points.

The coefficients obtained in these groups constitute the global soil flexibility matrix when positioned according to the degrees of freedom in the global coordinates (Figure 3.3). Group *A* non-diagonal sub-matrices are 3×3 square matrices with the flexibility coefficients  $A_1, \dots, A_9$ , and similarly, group *B* main diagonal sub-matrices are 3×3 square matrices with the flexibility coefficients  $B_1, \dots, B_9$ .

Elements of the flexibility matrix or the flexibility coefficients are represented by symbol  $f_{ab}^{mn}$ . In this notation, the subscripts *a* and *b* represent the point at which the displacement takes place and that where the load is applied, respectively. The superscripts  $m=i, j, k$  and  $n=i, j, k$  address the direction of the degrees of freedom in *x*, *y*, and *z* axes corresponding to the displacement and the load, respectively (Figure 3.4).

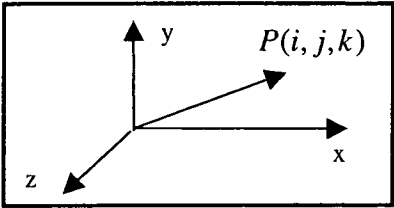


Figure 3.4 Orientation of direction vector for point  $P(i, j, k)$

When the soil flexibility matrix is assembled in this way it is symmetric about its main diagonal. Thus, by switching the sequence of both the superscripts and the subscripts simultaneously, the coefficients on both sides of the pivot can be defined (inter-change case). In a general form, this rule appears as follows (see Figure 3.5)

$$f_{ab}^{mn} = f_{ba}^{nm} \tag{3.8}$$

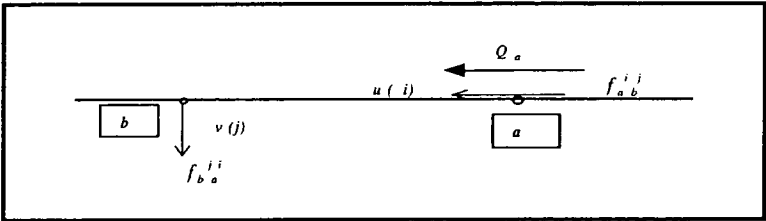


Figure 3.5 Flexibility coefficients associated with the horizontal and vertical degrees of freedom at point *b* due to horizontal force  $Q_a$  at point *a*.

The following notations are employed for loads and displacements:

- Load: Point loads and distributed loads are designated by  $P, Q, M$  and  $p, q, m$  (in upper and lower cases) respectively, with a subscript for the application point.
- Displacement: Symbols  $u, v$  and  $\varphi$  (in lower case) with subscripts related to the point where displacement evaluation takes place.

Arrangement of the coefficients in the soil flexibility matrix associated with two points  $a$  and  $b$  is presented in Figure 3.6.

$$[F]_{ab} = \begin{array}{c} \begin{array}{ccccc} & \overbrace{Q_a \quad P_a \quad M_a}^{\text{Node (a)}} & & \overbrace{Q_b \quad P_b \quad M_b}^{\text{Node (b)}} & \\ \begin{bmatrix} f_{aa}^{ii} & f_{aa}^{ij} & f_{aa}^{ik} \\ f_{aa}^{ji} & f_{aa}^{jj} & f_{aa}^{jk} \\ f_{aa}^{ki} & f_{aa}^{kj} & f_{aa}^{kk} \end{bmatrix} & \begin{bmatrix} f_{ab}^{ii} & f_{ab}^{ij} & f_{ab}^{ik} \\ f_{ab}^{ji} & f_{ab}^{jj} & f_{ab}^{jk} \\ f_{ab}^{ki} & f_{ab}^{kj} & f_{ab}^{kk} \end{bmatrix} & \begin{bmatrix} f_{ba}^{ii} & f_{ba}^{ij} & f_{ba}^{ik} \\ f_{ba}^{ji} & f_{ba}^{jj} & f_{ba}^{jk} \\ f_{ba}^{ki} & f_{ba}^{kj} & f_{ba}^{kk} \end{bmatrix} & \begin{bmatrix} f_{bb}^{ii} & f_{bb}^{ij} & f_{bb}^{ik} \\ f_{bb}^{ji} & f_{bb}^{jj} & f_{bb}^{jk} \\ f_{bb}^{ki} & f_{bb}^{kj} & f_{bb}^{kk} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} [F_{aa}] & [F_{ab}] \\ [F_{ba}] & [F_{bb}] \end{bmatrix} \end{array}$$

Figure 3.6 Illustration of flexibility coefficients in soil flexibility matrix  $[F]_{ab}$ .

The flexibility coefficients associated with supports according to their corresponding degrees of freedom in the global system forms the soil flexibility matrix. In the following sections, the flexibility coefficients associated with a homogenous soil with isotropic and cross-anisotropic properties are considered.

### 3.5.3 Flexibility Coefficients Associated with Two Distinct Points on a Homogeneous Isotropic Elastic Half-space

Force-displacement relations associated with two distinct points are directly used to develop the flexibility coefficients in this section. The loads considered include a horizontal and a vertical force ( $Q_b$  and  $P_b$ ) and a moment ( $M_b$ ) about the horizontal axis  $z$ .

For convenience in presentation, the following quantities are defined and used in this section and section 3.5.4:

$$\alpha = \frac{(1 - \nu_s^2)}{\pi E_s} \quad (3.9)$$

$$\beta = \frac{(1 + \nu_s)(1 - 2\nu_s)}{2\pi E_s} \quad (3.10)$$

$$\gamma = \frac{(1 + \nu_s)}{\pi E_s} \quad (3.11)$$

where  $\nu_s$  and  $E_s$  are Poisson's ratio and modulus of elasticity for soil.

### 3.5.3.1. Flexibility Coefficients Associated with Horizontal Force $Q_b$

Applying Cerruti force-displacement relations (Equations 2.8 and 2.9) results in the flexibility coefficients associated with the horizontal and the vertical displacements due to a horizontal force  $Q_b$  as follows:

$$u_a = \frac{(1 + \nu_s)}{\pi E_s R_{ab}} Q_b = \frac{\gamma}{R_{ab}} Q_b \quad (3.12)$$

$$v_a = \frac{(1 + \nu_s)(1 - 2\nu_s)}{2\pi E_s R_{ab}} Q_b = \frac{\beta}{R_{ab}} Q_b \quad (3.13)$$

Thus:

$$f_{ab}^{ii} = \frac{1 + \nu_s}{\pi E_s R_{ab}} = \frac{\gamma}{R_{ab}} \quad (3.14)$$

$$f_{ab}^{ji} = \frac{(1 + \nu_s)(1 - 2\nu_s)}{2\pi E_s R_{ab}} = \frac{\beta}{R_{ab}} \quad (3.15)$$

where  $R_{ab}$  is the distance between the point of the load application and that of the displacement concerned (i.e. points  $b$  and  $a$ ). In Figure 3.7, points  $a'$  and  $c'$  are the new locations of the footings after the horizontal load is applied.

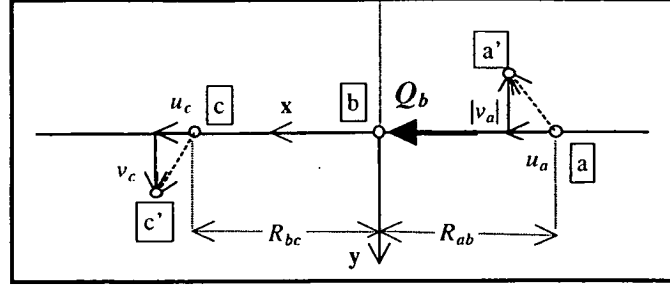


Figure 3.7. Profile of the surface displacement components due to horizontal surface force  $Q_b$ .

To determine the flexibility coefficients associated with the angle of rotation  $\varphi_a$  due to a horizontal load, the following approach is used.

Due to symmetry of the matrix in Figure 3.6,  $f_{ab}^{ki} = f_{ba}^{ik}$  and these two coefficients are interchangeable. Hence, horizontal displacement  $u_b$  due to moment  $M_a$  is calculated instead. For this purpose, moment  $M_a$  is replaced by a vertical couple  $P_1^*$  at distance  $\Delta$  (Figures 3.8 and 3.9).

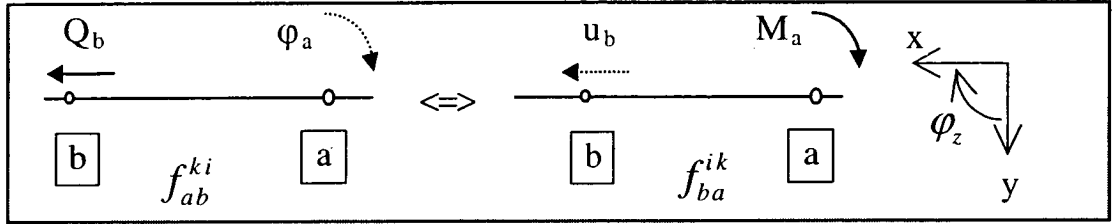


Figure 3.8 Equivalent situations in flexibility coefficients due to symmetry in the soil flexibility matrix.

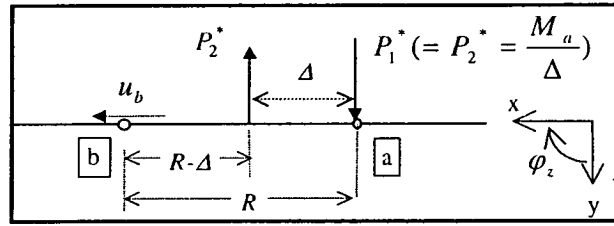


Figure 3.9 Resolution of moment  $M_a$  into a vertical couple acting on the surface of the medium.

By applying Equation 2.6 and letting  $\Delta \rightarrow 0$ , Equation 3.16 is obtained.

$$u_b = \lim_{\Delta \rightarrow 0} \frac{(1 + \nu_s)(1 - 2\nu_s)}{2\pi E_s \Delta} \left( -\frac{1}{R} + \frac{1}{R - \Delta} \right) M_a = \lim_{\Delta \rightarrow 0} \frac{\beta M_a}{R(R - \Delta)} \quad (3.16)$$

$$u_b = \frac{\beta M_a}{R^2} \quad (3.17)$$

For  $M_a = 1$ , the displacement  $u_b$  becomes the flexibility coefficient  $f_{ab}^{ki}$ . Hence

$$f_{ab}^{ki} = \frac{\beta}{R_{ab}^2} \quad (3.18)$$

The flexibility coefficients given by Equations 3.14, 3.15 and 3.18 correspond to positions  $A_1$ ,  $A_4$  and  $A_7$  in the flexibility matrix respectively (Figure 3.3).

### 3.5.3.2. Flexibility Coefficients Associated with Vertical Force $P_b$

As discussed earlier in Chapter 2, Boussinesq presented the horizontal and the vertical components of the displacement ( $u_a$  and  $v_a$ ) due to a vertical load in Equations 2.6 and 2.7.

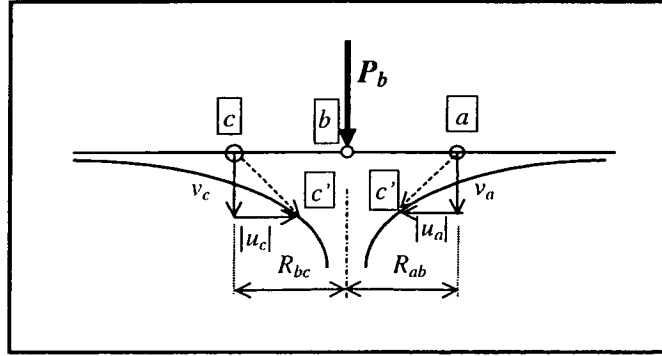


Figure 3.10 Profile of the surface horizontal and vertical displacements due to vertical force ( $P_b$ ).

The force displacement relations based on Boussinesq solution can be expressed as:

$$u_a = \frac{(1 + \nu_s)(1 - 2\nu_s)}{2\pi E_s R_{ab}} P_b \quad (3.19)$$

$$v_a = \frac{(1 - \nu_s^2)}{\pi E_s R_{ab}} P_b \quad (3.20)$$

where the quantities involved are either as shown in Figure 3.10 or as described earlier. Thus, the associated flexibility coefficients are:



$$f_{ab}^{ij} = \frac{(1 + \nu_s)(1 - 2\nu_s)}{2\pi E_s R_{ab}} = \frac{\beta}{R_{ab}} \quad (3.21)$$

$$f_{ab}^{jj} = \frac{(1 - \nu_s^2)}{\pi E_s R_{ab}} = \frac{\alpha}{R_{ab}} \quad (3.22)$$

Figure 3.11 shows the angle of rotation  $\varphi_a$  at point  $a$  caused by the vertical force  $P_b$  acting at point  $b$ .

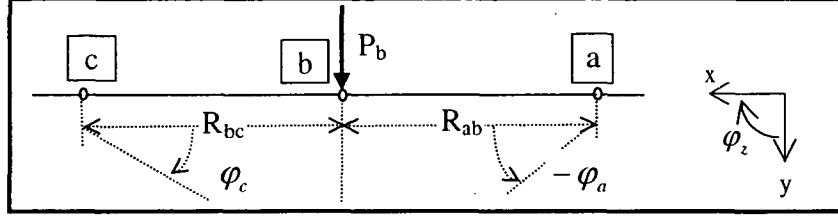


Figure 3.11 Rotations  $\varphi_a$  and  $\varphi_c$  due to the vertical point load  $P_b$ .

This angle of rotation can be found as the derivative of the vertical displacement at point  $a$ . Thus:

$$\varphi_a = \left. \frac{dv}{dr} \right|_{r=R_{ab}} = \left. \frac{d}{dr} \left( \frac{1 - \nu_s^2}{\pi E_s r} P_b \right) \right|_{r=R_{ab}} = -\frac{1 - \nu_s^2}{\pi E_s R_{ab}^2} P_b \quad (3.23)$$

or

$$\varphi_a = -\frac{\alpha}{R_{ab}^2} P_b \quad (3.24)$$

The corresponding flexibility coefficient ( $P_b = 1$ ), therefore, is:

$$f_{ab}^{kj} = -\frac{\alpha}{R_{ab}^2} \quad (3.25)$$

The flexibility coefficients obtained in Equations 3.21, 3.22 and 3.25 correspond to positions  $A_2$ ,  $A_3$  and  $A_8$ , respectively (Figure 3.3).

### 3.5.3.3. Flexibility Coefficients Associated with Moment $M_b$ about Horizontal Axis $z$

To obtain flexibility coefficients associated with moment  $M_b$ , the moment is replaced with a vertical couple as shown in Figure 3.12.

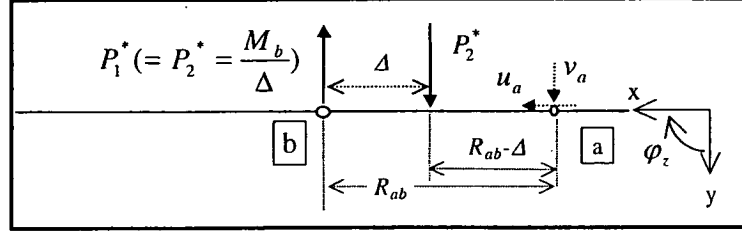


Figure 3.12 Replacement of moment  $M_b$  with a couple, and the vertical displacement.

Equation 3.19 is applied to determine the horizontal resultant displacement  $u_a$  due to forces  $P_1^*$  and  $P_2^*$ :

$$u_a = \lim_{\Delta \rightarrow 0} \frac{(1 + \nu_s)(1 - 2\nu_s)}{2\pi E_s \Delta} \left( -\frac{1}{R_{ab}} + \frac{1}{(R_{ab} - \Delta)} \right) M_b$$

$$u_a = \lim_{\Delta \rightarrow 0} \beta \left( \frac{1}{R_{ab}(R_{ab} - \Delta)} \right) M_b \quad (3.26)$$

or

$$u_a = \frac{\beta}{R_{ab}^2} M_b \quad (3.27)$$

Hence,

$$f_{ab}^{ik} = \frac{\beta}{R_{ab}^2} \quad (3.28)$$

The vertical displacement due to a couple is shown in Figure 3.12.

Vertical displacement  $v_a$  is determined with the aid of Equation 3.20 (Figure 3.12) as follows:

$$v_a = \lim_{\Delta \rightarrow 0} \frac{(1 - \nu_s^2)}{\pi E_s \Delta} \left( -\frac{1}{R_{ab}} + \frac{1}{(R_{ab} - \Delta)} \right) M_b = \lim_{\Delta \rightarrow 0} \frac{\alpha}{R_{ab}(R_{ab} - \Delta)} M_b \quad (3.29)$$

$$v_a = \frac{\alpha}{R_{ab}^2} M_b \quad (3.30)$$

Hence,

$$f_{ab}^{jk} = \frac{\alpha}{R_{ab}^2} \quad (3.31)$$

The angle of rotation at point  $a$  can be obtained by differentiating the vertical displacement (Equation 3.30) with respect to distance.

$$\varphi_a = -\frac{dv}{dr} \Big|_{r=R_{ab}} = -\frac{d}{dr} \left[ \frac{(1-\nu_s^2)}{\pi E_s r^2} M_b \right] \Big|_{r=R_{ab}} \quad (3.32)$$

As a result:

$$\varphi_a = \frac{2\alpha}{R_{ab}^3} M_b \quad (3.33)$$

Therefore, the corresponding flexibility coefficient is:

$$f_{ab}^{kk} = \frac{2\alpha}{R_{ab}^3} \quad (3.34)$$

The flexibility coefficients given by Equations 3.28, 3.31 and 3.34 correspond to positions  $A_3$ ,  $A_6$  and  $A_9$ , respectively (Figure 3.3).

### 3.5.4. Flexibility Coefficients Associated with Two Coincident Points on a Homogeneous Isotropic Elastic Half-space

If the point at which displacement is calculated coincides with the point of load application, the appropriate force-displacement relation is the same as the one for the case of two different points, except that in this case the relationship applies to distributed load. The flexibility coefficients associated with these relations constitute group B and are  $B_1, \dots, B_9$  (Figure 3.3).

### 3.5.4.1 Flexibility Coefficients Associated with Horizontal Force $Q_b$

In Figure 3.7, when points  $a$  and  $b$  are coincident, the Cerruti Equation 3.12 exhibits a singularity problem. As a result, the flexibility coefficient corresponding to the horizontal displacement due to such a load is evaluated as follows. It is assumed that the load is distributed over a small area in the vicinity of its application point. The displacement at that point is obtained by integrating Equation 3.12 over the area. It is better to choose a circular area, as it can easily lead the integration to a closed form (see Cheung and Zienkiewicz 1965, Melerski 1994, Izadnegahdar and Melerski 1997). For such a case, the associated flexibility coefficient with the horizontal displacement is given by:

$$f_{aa}^{ii} = f_{bb}^{ii} = \frac{(1 + \nu_s)(1 - 2\nu_s)}{\pi E_s r_0} = \frac{2\beta}{r_0} \quad (3.35)$$

where  $r_0$  is the radius of the assumed circle over which the load is distributed.

To obtain the corresponding flexibility coefficient for the vertical displacement, the total horizontal load acting at point  $b$  is divided into two equal forces which are applied on both sides in the vicinity of that point. Equation 3.13 gives the vertical displacements at both sides in the vicinity of this point (Figure 3.13).

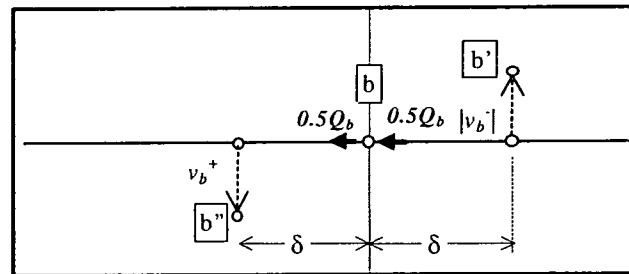


Figure 3.13 Total zero vertical displacement due to horizontal load  $Q_b$

These displacements are equal but in opposite directions. Hence, the total displacement, due to the continuity in surface deflection, is zero and the flexibility coefficient is:

$$f_{aa}^{ji} = f_{bb}^{ji} = 0 \quad (3.36)$$

To determine the flexibility coefficient associated with the angle of rotation due to horizontal load  $Q_b$ , the load is distributed uniformly over a small circular area, and Cerruti's equation (i.e. Equation 3.13) is applied to derive the vertical displacement. This rotation can be found as the derivative of the vertical displacement at point  $b$ . Thus:

$$\varphi_b = -\frac{dv}{dr}\bigg|_{r=r_0} = -\frac{d}{dr} \left[ \int^A \frac{(1+\nu_s)(1-2\nu_s)}{2\pi E_s r} dQ_b \right] \quad (3.37)$$

where  $dQ_b = q_b dA$ ,  $dA = r d\theta dr$  and  $q_b = \frac{Q_b}{\pi r_0^2}$ , then

$$\varphi_b = -2 \frac{(1+\nu_s)(1-2\nu_s)}{2\pi E_s} q_b \frac{d}{dr} \int_0^{r_0} \int_0^{2\pi} \frac{r d\theta dr}{r} = \frac{2\beta}{r_0^2} Q_b \quad (3.38)$$

Hence,

$$f_{bb}^{ki} = \frac{2\beta}{r_0^2} \quad (3.39)$$

The flexibility coefficients given by Equations 3.35, 3.36 and 3.39 correspond to positions  $B_1$ ,  $B_4$  and  $B_7$ , respectively (Figure 3.3).

#### 3.5.4.2. Flexibility Coefficients Associated with Vertical Force $P_b$

The horizontal displacement under vertical load  $P_b$  is obtained by considering the axial symmetry about the line of force action. Due to symmetry, such a displacement is zero and the associated flexibility coefficient is:

$$f_{aa}^{ij} = f_{bb}^{ij} = 0 \quad (3.40)$$

To obtain the flexibility coefficient associated with the vertical displacement under a vertical point load when the distance ( $R_{ab}$ ) tends to zero, direct application of Equation 3.20 leads to singularity. To overcome this singularity problem the vertical load is distributed over a small circle containing that point. Integration of the vertical

displacement due to the load over the infinitesimal area facilitates the calculation and leads to a finite value for the total vertical displacement (see Appendix A1) (Cheung and Zienkiewicz 1965). The flexibility coefficient is then given as:

$$f_{aa}^{jj} = f_{bb}^{jj} = \frac{2(1-\nu_s^2)}{\pi E_s r_0} = \frac{2\alpha}{r_0} \quad (3.41)$$

where  $r_0$  is the radius of the assumed load distribution circle, and the remaining quantities were defined in section 2.3.2.

Cheung and Zienkiewicz (1965) investigated load distribution over a foundation with different shapes such as a square and a rectangle with different aspect ratio. In that study, a rectangle  $a \times b$  was exposed to a uniform load and the Boussinesq force-displacement relationship was integrated over the area to obtain the vertical displacement under the rectangle.

To get the angle of rotation under a vertical load, symmetry of the load about the line of action is considered. Since the displacements due to applied force are symmetric the slope under the force must be zero. Thus:

$$f_{bb}^{kj} = 0 \quad (3.42)$$

The flexibility coefficients given by Equations 3.40, 3.41 and 3.42 correspond to positions  $B_2$ ,  $B_5$  and  $B_8$ , respectively (Figure 3.3).

#### **3.5.4.3. Flexibility Coefficients Associated with Moment $M_b$ about Horizontal Axis $z$**

The flexibility coefficients ( $B_3$  and  $B_6$ ) associated with the horizontal and vertical displacements are symmetrical with those at positions  $B_7$  and  $B_8$ , which are obtained from Equations 3.39 and 3.42. Hence, only the flexibility coefficient that corresponds to the angle of rotation  $\phi_b$  is developed here.

Applying the method used for non-coincident points (section 3.5.3.3), as the distance  $R_{ab}$  tends to zero would cause singularity in Equation 3.33. Therefore, the moment  $M_b$  is replaced with a vertical linearly distributed load over an assumed circle ( $r = r_0$ , Figure 3.14). The total vertical load over both compression and tension half-circles are equal but in opposite directions. Hence, the integration of the vertical displacement due to the infinitesimal load over the area gives the vertical displacement and the derivative of this displacement with respect to the distance results in the angular change at that point.

The Boussinesq expression (i.e. Equation 3.20) is applied, and the total vertical displacement due to the distributed load over half of the area is obtained.

$$v_b = \int \frac{(1-\nu_s^2) p_b dA}{\pi E_s r} \quad (3.43)$$

where  $p_b = \frac{p_{\max} r \cos \theta}{r_0}$  and  $dA = r d\theta \times dr$  are the load and the corresponding infinitesimal area which the load acts upon, respectively. Substituting these quantities in Equation 3.42, the vertical displacement is obtained.

$$v_a = \int \frac{(1-\nu_s^2)}{\pi E_s} \times \frac{p_{\max} r \cos \theta}{r_0} \times \frac{r d\theta dr}{r} = \frac{2 \alpha p_{\max}}{r_0} \int_0^{r_0} \int_0^{\pi/2} r \cos \theta d\theta dr \quad (3.44)$$

The angle of rotation is obtained as a derivative of the vertical displacement with respect to distance  $r$  as follows:

$$\varphi_b = \left. \frac{dv_a}{dr} \right|_{r=r_0} = \frac{2 \alpha p_{\max} r}{r_0} \int_0^{\pi/2} \cos \theta d\theta = \frac{2 \alpha p_{\max}}{r_0} r \sin \theta \Big|_0^{\pi/2} \quad (3.45)$$

$$\varphi_b = \frac{2 \alpha p_{\max} r}{r_0} \quad (3.46)$$

Hence, the angle of rotation at  $r = r_0$  becomes:

$$\varphi_b = 2 \alpha p_{\max} \quad (3.47)$$

The relation between moment  $M_b$  and a linearly distributed load over a circle (Figure 3.14) is obtained as below.

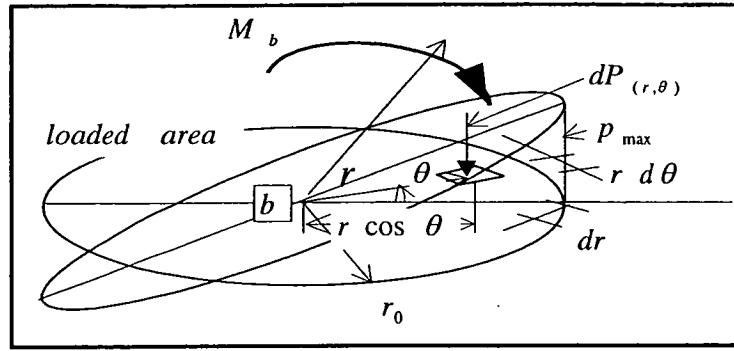


Figure 3.14 Load distribution over a circle on the soil replacing moment  $M_b$

The moment produced by such a vertical load over an infinitesimal area is as follows:

$$dM_b = (r \cos \theta) dP_b = r \cos \theta p_b dA \quad (3.48)$$

where  $dP_b$  is the vertical force acting on the small area and  $r \cos \theta$  is (the projection of distance  $r$  on  $x$  direction) the distance of that force from the  $z$  axis. The vertical linear load distribution is given by:

$$p_b(r, \theta) = \frac{p_{\max}}{r_0} r \cos \theta \quad (3.49)$$

The total moment due to such a load distribution over the entire area is calculated by integration.

$$M_b = \int_A dM_b = 4 \int_0^{r_0} \int_0^{\pi/2} r^2 \cos \theta p(r, \theta) dr d\theta = \frac{\pi}{4} p_{\max} r_0^3 \quad (3.50)$$

Thus, the following relation between the maximum vertical load representing a moment about horizontal axis and this moment is obtained.

$$p_{\max} = \frac{4}{\pi r_0^3} M_b \quad (3.51)$$

Substituting  $p_{\max}$  from Equation 3.51, the angle of rotation is obtained as follows:



$$\varphi_b = 2\alpha p_{\max} = \frac{8\alpha}{\pi r_0^3} M_b \quad (3.52)$$

Hence, the flexibility coefficient is:

$$f_{bb}^{kk} = \frac{8\alpha}{\pi r_0^3} \quad (3.53)$$

This coefficient corresponds to position  $B_9$  in the flexibility matrix (Figure 3.3).

### 3.5.5. Flexibility Coefficients Associated with Two Distinct Points on a Homogenous Cross-anisotropic Elastic Half-space

In this section, the flexibility coefficients associated with the displacements due to the application of loads to a medium with cross-anisotropic properties are developed. The development is based on the solutions discussed in the previous chapter. As mentioned in Chapter 2, when points of load application and displacement evaluation are different, the solutions from Gerrard and Wardle (1973) are applied, and the flexibility coefficients obtained.

The force-displacement relations due to three types of loads are discussed: horizontal force  $Q_b$ , vertical force  $P_b$ , and moment  $M_b$  about the horizontal axis  $z$ , and the flexibility coefficients are obtained. The relationships for this soil model, as discussed in Chapter 2, are used while the sign convention for global Cartesian coordinates (Figure 3.1) is adopted.

As mentioned previously in section 2.4, the quantity  $\beta^2$  is defined based on the cross-anisotropic properties. Hence, three cases are taken into account (i.e.:  $\beta^2 > 0$ ,  $\beta^2 < 0$  and  $\beta^2 = 0$ ) for individual loads (i.e. horizontal force  $Q_b$ , vertical force  $P_b$ , and moment  $M_b$  about the horizontal axis  $z$ ). In these expressions, the elastic parameters as well as the integration coefficients are introduced in Appendices A2.1.2 and A2.1.2.2.

### 3.5.5.1. Flexibility Coefficients Associated with the Displacements due to Horizontal Load $Q_b$

Displacement components  $u_a$ ,  $v_a$  and  $\varphi_a$  are due to a horizontal point load  $Q_b$  (Gerrard and Wardle 1973) (Figure 2.4a), and the corresponding flexibility coefficients for different cases of  $\beta^2$  are given as follows:

- When  $\beta^2 > 0$ :

$$u_a = \frac{h_{10}}{2\pi R_{ab}} Q_b = f_{ab}^{ii} Q_b \quad (3.54)$$

$$v_a = \frac{h_1 - h_2}{2\pi R_{ab}} Q_b = f_{ab}^{jj} Q_b \quad (3.55)$$

$$\varphi_a = \frac{h_1 - h_2}{2\pi R_{ab}^2} Q_b = f_{ab}^{ki} Q_b \quad (3.56)$$

where  $h_1$ ,  $h_2$  and  $h_{10}$  quantities are integral constants that are defined in Appendix A2.1.2.2.

- For  $\beta^2 < 0$  then:

$$u_a = \frac{j_{11}}{2\pi R_{ab}} Q_b = f_{ab}^{ii} Q_b \quad (3.57)$$

$$v_a = \frac{(r_{ab} - \alpha) j_1}{2\pi R_{ab}^2} Q_b = f_{ab}^{ji} Q_b \quad (3.58)$$

$$\varphi_a = \frac{j_1 (r_{ab} - 2\alpha)}{2\pi R_{ab}^3} Q_b = f_{ab}^{ki} Q_b \quad (3.59)$$

where  $j_1$  and  $j_{11}$  are integral constants as defined in Appendix A2.1.2.2

- If  $\beta^2 = 0$  then:

$$u_a = \frac{t_{10}}{2\pi R_{ab}} Q_b = f_{ab}^{ii} Q_b \quad (3.60)$$

$$v_a = \frac{t_1}{2\pi R_{ab}} Q_b = f_{ab}^{ji} Q_b \quad (3.61)$$

$$\varphi_a = \frac{t_1}{2\pi R_{ab}^2} Q_b = f_{ab}^{ki} Q_b \quad (3.62)$$

where  $t_1$  and  $t_{10}$  are integral constants as defined in Appendix A2.1.2.2

### 3.5.5.2. Flexibility Coefficients Associated with the Displacements due to Vertical Load $P_b$

Displacement components  $u_a$ ,  $v_a$  and  $\varphi_a$  are due to the application of vertical point load  $P_b$  (Figure 2.4b) (Gerrard and Wardle 1973), and the corresponding flexibility coefficients for different cases of  $\beta^2$  are shown below.

- When  $\beta^2 > 0$ :

$$u_a = \frac{g_3 - g_4}{2\pi R_{ab}} P_b = f_{ab}^{ij} P_b \quad (3.63)$$

$$v_a = \frac{g_1 - g_2}{2\pi R_{ab}} P_b = f_{ab}^{jj} P_b \quad (3.64)$$

$$\varphi_a = \frac{g_1 - g_2}{2\pi R_{ab}^2} P_b = f_{ab}^{kj} P_b \quad (3.65)$$

where  $g_1$ ,  $g_2$ ,  $g_3$  and  $g_4$  are integral constants as defined in Appendix A2.1.2.2

- For  $\beta^2 < 0$ :

$$u_a = \frac{(\alpha - r_{ab}) i_3}{2\pi R_{ab}^2} P_b = f_{ab}^{ij} P_b \quad (3.66)$$

$$v_a = \frac{i_1}{2\pi R_{ab}} P_b = f_{ab}^{jj} P_b \quad (3.67)$$

$$\varphi_a = \frac{i_1}{2\pi R_{ab}^2} P_b = f_{ab}^{kj} P_b \quad (3.68)$$

where  $i_1$  and  $i_3$  are integral constants as defined in Appendix A2.1.2.2

- If  $\beta^2 = 0$ :

$$u_a = \frac{-s_3}{2\pi R_{ab}} P_b = f_{ab}^{ij} P_b \quad (3.69)$$

$$v_a = \frac{s_1}{2\pi R_{ab}} P_b = f_{ab}^{jj} P_b \quad (3.70)$$

$$\varphi_a = \frac{s_1}{2\pi R_{ab}^2} P_b = f_{ab}^{kj} P_b \quad (3.71)$$

where  $s_1$  and  $s_3$  are integral constants as defined in Appendix A2.1.2.2.

### 3.5.5.3. Flexibility Coefficients Associated with the Displacements due to Moment $M_b$ about Horizontal Axis $z$

The horizontal, vertical and angular displacements  $u_a$ ,  $v_a$  and  $\varphi_a$  resulting from the moment  $M_b$  (Gerrard and Wardle 1973) at point  $a$  on the surface of a cross-anisotropic elastic half-space are described below (see Figure 2.4c). The flexibility coefficients associated with the displacements for different quantities of  $\beta^2$  are as follows:

- When  $\beta^2 > 0$  then:

$$u_a = \frac{g_4 - g_3}{2\pi R_{ab}^2} M_b = f_{ab}^{ik} M_b \quad (3.72)$$

$$v_a = \frac{g_2 - g_1}{2\pi R_{ab}^2} M_b = f_{ab}^{jk} M_b \quad (3.73)$$

$$\varphi_a = \frac{g_2 - g_1}{2\pi R_{ab}^3} M_b = f_{ab}^{kk} M_b \quad (3.74)$$

where  $g_1, g_2, g_3$  and  $g_4$  are integral constants as defined in Appendix A2.1.2.2

- For  $\beta^2 < 0$ :

$$u_a = \frac{i_3}{2\pi R_{ab}^2} M_b = f_{ab}^{ik} M_b \quad (3.75)$$

$$v_a = \frac{-i_1}{2\pi R_{ab}^2} M_b = f_{ab}^{jk} M_b \quad (3.76)$$

$$\varphi_a = \frac{-i_1}{2\pi R_{ab}^3} M_b = f_{ab}^{kk} M_b \quad (3.77)$$

where  $i_1$  and  $i_3$  are integral constants as defined in Appendix A2.1.2.2

- If  $\beta^2 = 0$ :

$$u_a = \frac{s_3}{\pi R_{ab}^2} M_b = f_{ab}^{ik} M_b \quad (3.78)$$

$$v_a = \frac{-s_1}{\pi R_{ab}^2} M_b = f_{ab}^{jk} M_b \quad (3.79)$$

$$\varphi_a = \frac{-s_1}{\pi R_{ab}^3} M_b = f_{ab}^{kk} M_b \quad (3.80)$$

where  $s_1$  and  $s_3$  are integral constants as defined in Appendix A2.1.2.2

### 3.5.6. Flexibility Coefficients Associated with Two Coincident Points on a Homogenous Cross-anisotropic Elastic Half-space

When the points of load application and displacement evaluation are coincident, Gerrard and Harrison (1970a) solutions are applied to obtain the flexibility coefficients.

As discussed in the previous chapter, to derive the force-displacement relations, the horizontal and vertical concentrated loads are substituted with distributed loads over

a circle, which produce horizontal and vertical uniform stresses respectively ( $q_b$ ,  $p_b$ ). The moment  $M_b$  is replaced with a linearly distributed load over a circle with its maximum value  $p_{b, \max}$  appearing at  $r = r_0$ .

### 3.5.6.1. Flexibility Coefficients Associated with the Displacements due to Horizontal Force $Q_b$

As shown earlier in Figure 2.5a, the horizontal force  $Q_b$  is applied at point  $b$  and the horizontal, vertical and angular components of the displacement at point  $b$  are determined (Gerrard and Harrison 1970a). Corresponding flexibility coefficients for different cases of  $\beta^2$  are presented below.

The uniformly distributed tangential load equivalent to a concentrated horizontal force is given by:

$$q = \frac{1}{\pi r_0^2} Q_b \quad (3.81)$$

where  $r_0$  is the radius of the load area

- When  $\beta^2 > 0$ , then

$$u_b = -\frac{(h_3 - h_4 - h_{10})}{2\pi r_0} Q_b = f_{bb}^{ii} Q_b \quad (3.82)$$

$$v_b = 0 = f_{bb}^{ji} Q_b \quad (3.83)$$

$$\varphi_b = \frac{(h_2 - h_1)}{2\pi r_0^2} Q_b = f_{bb}^{ki} Q_b \quad (3.84)$$

where  $h_1, h_2, h_3, h_4$  and  $h_{10}$  are integral constants as defined in Appendix A2.1.2.2.

- For  $\beta^2 < 0$

$$u_b = -\frac{(j_3 - j_{11})}{2\pi r_0} Q_b = f_{bb}^{ii} Q_b \quad (3.85)$$

$$v_b = 0 = f_{bb}^{ji} Q_b \quad (3.86)$$

$$\varphi_b = \frac{-j_1}{2\pi r_0^2} Q_b = f_{bb}^{ki} Q_b \quad (3.87)$$

where  $j_1, j_3$  and  $j_{11}$  are integral constants and are defined in Appendix A2.1.2.2.

- If  $\beta^2 = 0$

$$u_b = \frac{(t_{10} + t_3)}{2\pi r_0} Q_b = f_{bb}^{ii} Q_b \quad (3.88)$$

$$v_b = 0 = f_{bb}^{ji} Q_b \quad (3.89)$$

$$\varphi_b = \frac{-t_1}{2\pi r_0^2} Q_b = f_{bb}^{ki} Q_b \quad (3.90)$$

where  $t_1, t_3$  and  $j_{10}$  are integral constants as defined in Appendix A2.1.2.2.

### 3.5.6.2. Flexibility Coefficients Associated with the Displacements due to Vertical Force $P_b$

The horizontal, vertical and angular components of the displacement at point  $b$  due to the vertical force applied at the same point (Gerrard and Harrison 1970a), and their corresponding flexibility coefficients for different cases of  $\beta^2$  are presented below (Figure 2.5b):

The uniform vertical load equivalent to that of a concentrated vertical force is as follows:

$$p = \frac{1}{\pi r_0^2} P_b \quad (3.91)$$

- For  $\beta^2 > 0$ , then

$$u_b = 0 = f_{bb}^{ij} P_b \quad (3.92)$$

$$v_b = \frac{(g_1 - g_2)}{\pi r_0} P_b = f_{bb}^{jj} P_b \quad (3.93)$$

$$\varphi_b = 0 = f_{bb}^{kj} P_b \quad (3.94)$$

where  $g_1$  and  $g_2$  are integral constants as defined in Appendix A2.1.2.2.

- For  $\beta^2 < 0$ :

$$u_b = 0 = f_{bb}^{ij} P_b \quad (3.95)$$

$$v_b = \frac{i_1}{\pi r_0} P_b = f_{bb}^{jj} P_b \quad (3.96)$$

$$\varphi_b = 0 = f_{bb}^{kj} P_b \quad (3.97)$$

where  $i_1$  is integral constant as defined in Appendix A2.1.2.2.

- For  $\beta^2 = 0$ , then

$$u_b = \frac{s_3}{\pi r_0} P_b = f_{bb}^{ij} P_b \quad (3.98)$$

$$v_b = \frac{s_1}{\pi r_0} = f_{bb}^{jj} P_b \quad (3.99)$$

$$\varphi_b = 0 = f_{bb}^{kj} P_b \quad (3.100)$$

where  $s_1$  and  $s_3$  are integral constants as defined in Appendix A2.1.2.2.



### 3.5.6.3. Flexibility Coefficients Associated with the Displacements due to Moment $M_b$ about Horizontal Axis $z$

With reference to Figure 2.5, the horizontal, vertical and angular displacements at point  $b$  due to the moment  $M_b$  applied at the same point (Gerrard and Harrison 1970a), and their corresponding flexibility coefficients for different cases of  $\beta^2$  are presented below.

- As shown earlier in section 3.5.4.3, the moment  $M_b$  is replaced by its equivalent in terms of a linearly distributed load over a circle (Figure 3.15). The moment produced by the load distributed over an infinitesimal area  $dA = r d\theta \times dr$  is obtained as

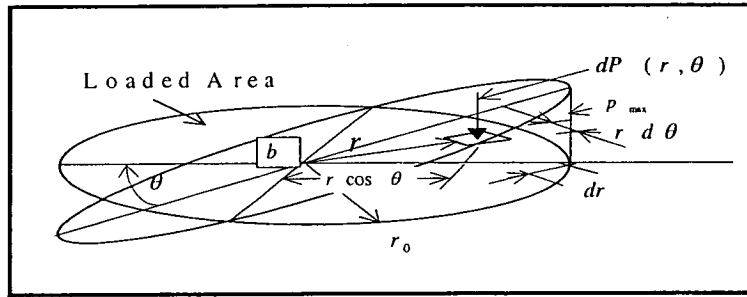


Figure 3.15 Load distribution over a circle on the soil replacing moment  $M_b$ .

$$dM = (r \cos \theta).dP = r \cos \theta.P(r, \theta).dA \quad (3.101)$$

where  $dP$  is the vertical force acting on the small area and  $r \cos \theta$  is the distance of that force from the  $z$  axis (the projection of distance  $r$  on  $x$  direction). The vertical linear load distribution is given as follows:

$$p(r, \theta) = \frac{P_{\max}}{r_0} r \cos \theta \quad (3.102)$$

The total moment due to such a load distribution over the entire area results from integration.

$$M_b = \int_0^A dM_b = 4 \int_0^{r_0} \int_0^{\pi/2} r^2 \cos \theta p(r, \theta) dr d\theta = \frac{\pi}{4} p_{\max} r_0^3 \quad (3.103)$$

Thus, the maximum vertical stress caused by this concentrated moment is

$$p_{\max} = \frac{4}{\pi r_0^3} M_b \quad (3.104)$$

The resulting displacements and flexibility coefficients are:

- When  $\beta^2 > 0$ , then:

$$u_b = \frac{(g_3 - g_4)}{\pi r_0^2} M_b = f_{bb}^{ik} M_b \quad (3.105)$$

$$v_b = 0 = f_{bb}^{jk} M_b \quad (3.106)$$

$$\varphi_b = \frac{2(g_1 - g_2)}{\pi r_0^3} M_b = f_{bb}^{kk} M_b \quad (3.107)$$

where  $g_1, g_2, g_3$  and  $g_4$  are integral constants as defined in Appendix A2.1.2.2.

- For  $\beta^2 < 0$ , then:

$$u_b = \frac{-i_3}{\pi r_0^2} M_b = f_{bb}^{ik} M_b \quad (3.108)$$

$$v_b = 0 = f_{bb}^{jk} M_b \quad (3.109)$$

$$\varphi_b = \frac{2i_1}{\pi r_0^3} M_b = f_{bb}^{kk} M_b \quad (3.110)$$

where  $i_1$  and  $i_3$  are integral constants as defined in Appendix A2.1.2.2.

If  $\beta^2 = 0$ , then:

$$u_b = \frac{-s_3}{\pi r_0^2} M_b = f_{bb}^{ik} M_b \quad (3.111)$$

$$v_b = 0 = f_{bb}^{jk} M_b \quad (3.112)$$

$$\varphi_b = \frac{2 s_1}{\pi r_0^3} M_b = f_{bb}^{kk} M_b \quad (3.113)$$

where  $s_1$  and  $s_3$  are integral constants as defined in Appendix A2.1.2.2.

### 3.6. CONCLUDING REMARKS TO CHAPTER

In this chapter, the combined and integrated system of soil-structure was introduced and the Direct Stiffness Matrix Method applied to analyse the system. The structure stiffness matrix for the soil-structure system was obtained. The force-displacement relationships for application of three different loads (horizontal force  $Q_b$ , vertical force  $P_b$ , and moment  $M_b$  about the horizontal axis  $z$ ) to the soil media with isotropic and cross-anisotropic properties were obtained. The flexibility coefficients associated with two types of soils that is, isotropic elastic half-space and cross-anisotropic elastic half-space were developed.

# CHAPTER

# FOUR

*SOFTWARE OF DEVELOPMENT*

*Elastostatic Interaction Analysis of Frames  
Resting on Homogeneous Elastic Half-space*

# CHAPTER 4

## SOFTWARE DEVELOPMENT

### 4.1. INTRODUCTION

The computer method discussed previously in section 3.3 for solving the elastic problem of the interaction between plane frame structures and a homogeneous elastic half-space medium is employed in a computer program that is introduced in this chapter. The data input and computation handling are facilitated by the use of windows, pull-down menus and dialogue boxes. The program requires minimal data and provides important structural response fields for design with modest computation time. Attention has been paid to providing the required checks to validate the input data/file and avoid quitting the program execution. Nevertheless, it is important that the user checks the analysis requirements, correctness of the data and its format in advance.

Two computer program modules were written: the first to enter the structural and loading data ( *DATAENTR* ), and the second to analyse the structure ( *ANALYSIS* ). These two programs are linked through a batch program ( *Genforms.bat* ). These two modules together with the batch program constitute an integrated program "Soil And Structure Interaction Analysis Package (*SASIAP* 1.0)" that the author developed as a tool for this research. Both modules are written in *Turbo Pascal* language (6.0 and 7.0 Borland 1990a -d, 1992a -d, O'Brien 1991, Palmer 1991). The former is enhanced with *Turbo Vision* (2.0) tools. Flow charts of Figures 4.1 to 4.3 provide an overview of *SASIAP*.

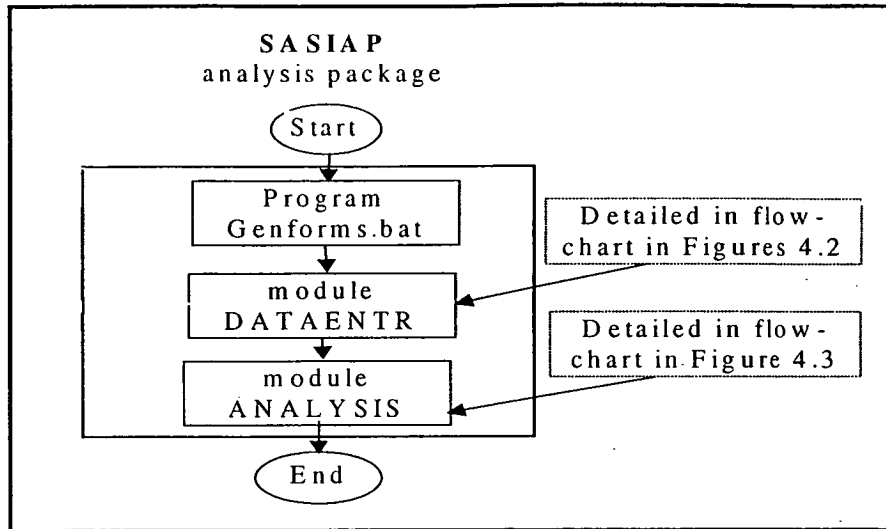


Figure 4.1 Overview of *SASIAP* analysis package.

## 4.2. MODULE "DATAENTR"

The module *DATAENTR* has been developed as a *DOS* application with an executable file size of approximately 300 Kbytes. This module is a program that handles the structural and load data. At different stages, dialogue boxes with appropriate configurations are provided to facilitate data entry and editing of the data for structure and loads. This requirement was a significant factor in the selection of the language in which the program was written. Turbo Vision was the language chosen to write *DATAENTR* as Turbo Vision allows all the required features such as pull down menus, dialogue boxes, buttons and message boxes, to be incorporated into the program. A user's manual was prepared to help the user with *SASIAP*, although the program design is such that users familiar with structural data requirements should have no difficulties. The program manual and the code listings are provided in detail in Appendix 4.

The program *DATAENTR* manages the data and is supported by a sub-program and two groups of computer programming units such as Dos, Objects, Drivers, Memory, Views (standard units), and *MlistDlg*, *Mforms* (modified units), as described below.

The first group consists of some standard units from Turbo Pascal 6.0 and 7.0, and the second group constitutes the modified standard units in conjunction with the standard units and new units that are prepared for individual needs.

There are two sets of dialogue boxes that manage the data. The first set facilitates the entry of the following structural data:

- Nodal Geometry
- Material Properties
- Cross Section Properties
- Member Data
- Nodal Restraints

The second set is designed to handle the structure loading data. The load cases catered for are:

- Nodal Load
- Nodal Prescribed Displacement Load
- Member Point Load
- Member Uniformly Distributed Load
- Member Trapezoidal Load

Figure 4.2 presents the flow chart for the module *DATAENTR*, and the sub-module *Genform* (a) is illustrated in Figure 4.3. A listing of the main program, sub-program and the referred programming units is included in Appendix A4.6.

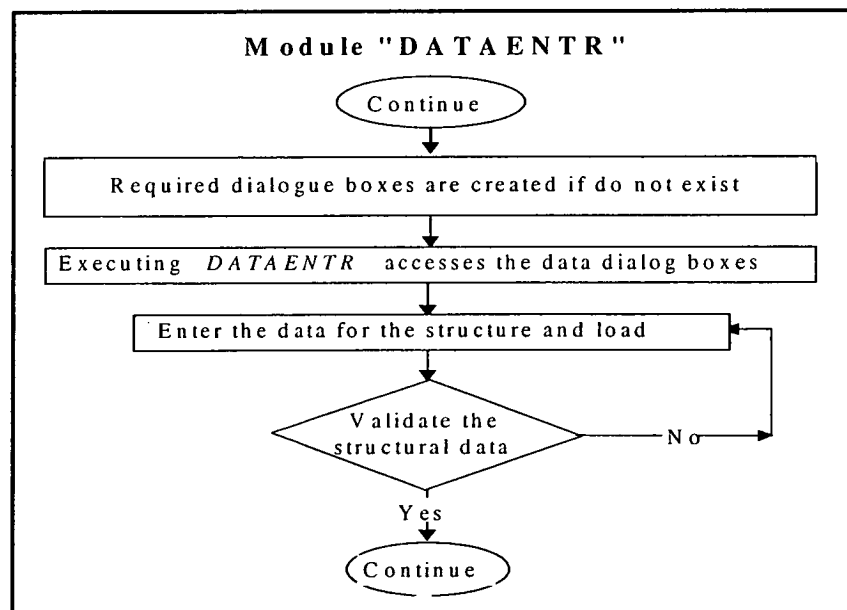


Figure 4.2 Flow chart for module "*DATAENTR*".

When *DATENTR* is run, pre-configured blank dialogue boxes are generated by a batch program known as *Genforms*. Through individual procedures, the data as well as loading information are entered and saved in two formats: *Text* format for the user view, and *Binary* format for the program access. This data becomes available for the whole analysis procedure. At this point, the user can retrieve all the data for any additional (supplementary) analyses required.

### 4.3. MODULE "ANALYSIS"

The module *ANALYSIS* was developed as a *DOS* application with an executable size of 126 Kbytes. This module searches in the working directory for the required programming units and the required data files. In addition to some standard Pascal 6.0 programming units, there are three other units, namely: *Comp* to accommodate variables and constants, *Build* mainly for checking data consistency and *SoilMode* for the soil consideration in the analysis. The execution task is controlled by appropriate responses received from the user interface via the keyboard. The program considers several types of nodal restraints such as: fully fixed (infinitely rigid), linear elastic-restrained which has some spring coefficient and is free against any movement. Soil models that can be considered are elastic in 3D space with isotropic and cross-anisotropic properties. The flow chart for the module *ANALYSIS*, the soil and structure sections and the analysis procedure in an abbreviated form are provided in Figures 4.3 to 4.6.

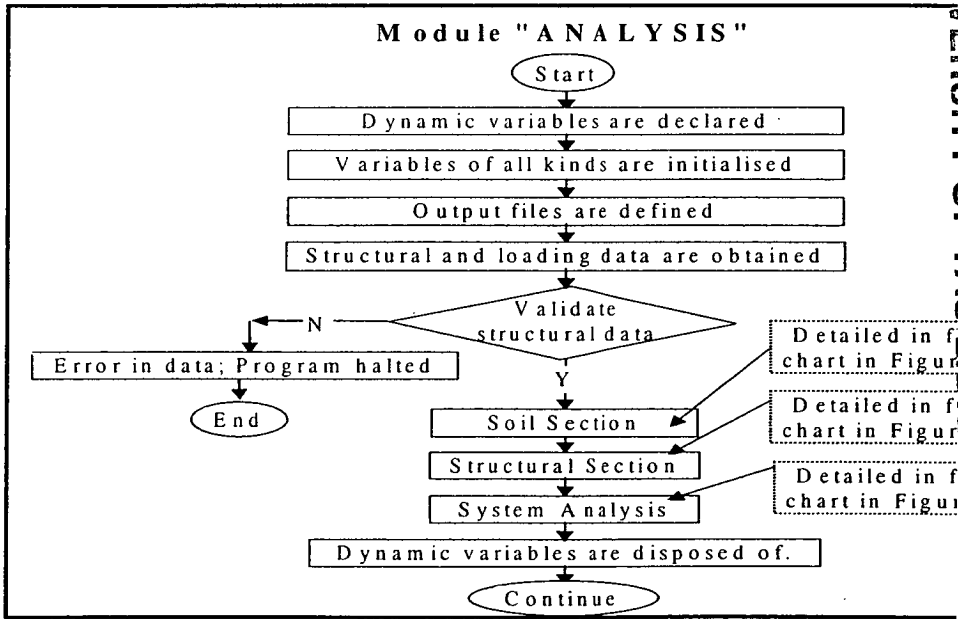


Figure 4.3 Flow chart for module "ANALYSIS".



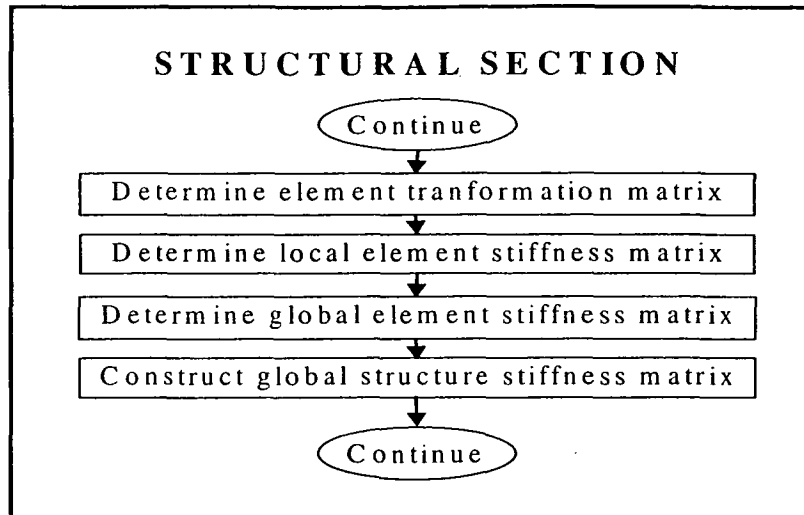


Figure 4.5 Flow chart for structure consideration in analysis procedure (see also Figure 4.3).

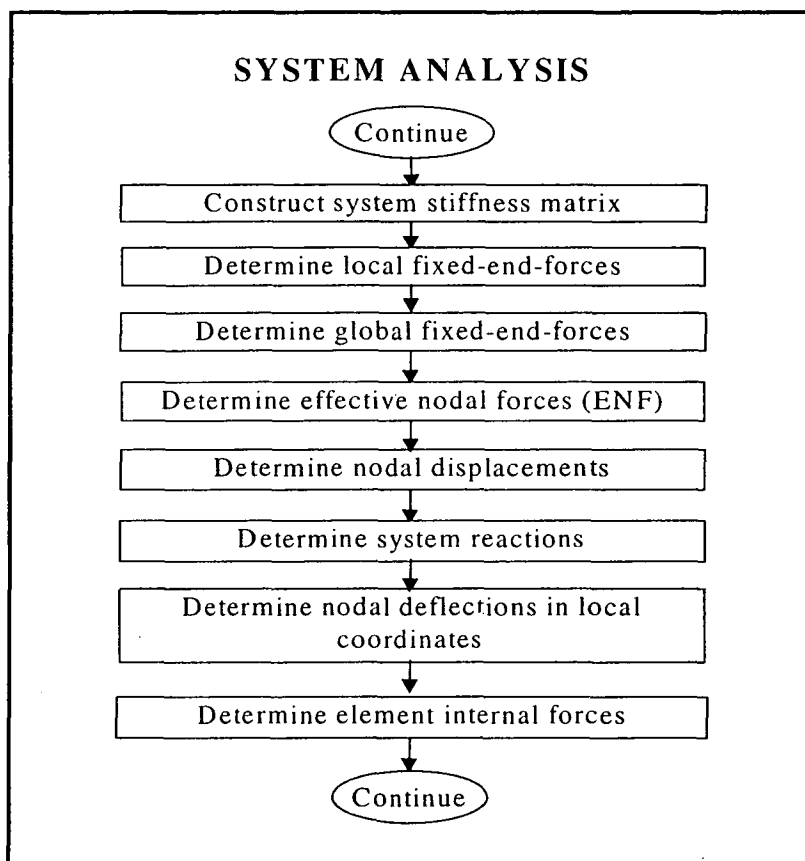


Figure 4.6 Flow chart for the system analysis (see also Figure 4.3).

As mentioned earlier, the program first validates the data and then the restraint conditions are selected through the user interface. Any of the following options becomes available for analysis for an elastic soil:

- A single plane frame structure with no contribution from the effect of the neighbouring frames or the adjacent support nodes in the same frame (i.e. when

*DiagonalAlone* is chosen). In this case, only the effect of support reactions under isolated individual nodes of a single frame on the soil in the structural analysis are evaluated.

- A single plane frame structure, incorporating the effect of the neighbouring support nodes of the same frame in that plane (i.e. when *DiagonalAlone* is not selected). This option is available for the study of not only of the individual support reactions of the single frame, but also the interaction of the other support nodes from the same frame, on each other, in the structural analysis.
- The effect of nodes from neighbouring parallel frames is considered (i.e. options: *DiagonalAlone* is not accepted, but *3D-extension* is selected). This option is available to study the interaction amongst the support nodes of the same frame. The effect of other frames, which are assumed to be in parallel and to be related to the frame concerned through the soil medium (only in a direction perpendicular to the frame) is also taken into account.

#### 4.4. OUTPUT OF ANALYSIS

When the program *ANALYSIS* is executed the output is written into text files in three different forms. One form gives the full information of the structure including nodes, elements, loads, internal forces and the nodal displacements and soil information. The second form prepares only a listing of the structural elements, the internal forces and the nodal displacements. The final form provides a summary of the reactions.

#### 4.5. CONCLUDING REMARKS TO CHAPTER

In this chapter, the methodology discussed in the previous chapter was applied to construct the computer program. This approach is shown through use of appropriate flow charts and the program manual is prepared to help user with data entry. The program *ANALYSIS* can be used to analyse plane frames with contribution of two assigned soil types (isotropic and cross-anisotropic elastic half-space) at support nodes. Finally, output files are obtained in various forms that are tailored to provide specific information on structural properties, internal forces, nodal displacements and reactions at superstructure supports.

# CHAPTER

# FIVE

*EXAMPLES OF ANALYSIS APPLICATIONS*

*Elastostatic Interaction Analysis of Frames  
Resting on Homogeneous Elastic Half-space*

## CHAPTER 5

### EXAMPLES OF ANALYSIS APPLICATIONS

#### 5.1. INTRODUCTION

In this chapter, two examples of a plane frame are analysed using the computer program developed in the study and described in the previous chapter. The soil is first modelled as an elastic half-space with isotropic and subsequently with cross-anisotropic properties. A typical frame with pinned support condition is considered for this study. The example was analysed both as a single frame and as a member of a series of interacting frames.

#### 5.2. HINGE-SUPPORTED FRAME AND SOIL MODEL APPLIED

In the examples, the frame is assumed to be pinned at its supports. Hence, the nodal supports are restrained against free translations in the horizontal and vertical directions.

The following cases are studied:

- a) A single plane frame (Figure 5.1a) analysis where no interaction between the neighbouring supports is considered. In tables 5.1a to 5.1d, this option is represented by letter “*D*” as in the last character of the file name which refers to *diagonal blocks* in the soil flexibility matrix.
- b) A single plane frame (Figure 5.1b) is considered with soil interaction between its supports. This option is represented by letter “*P*” as in the last character of the file name which stands for *plane frame only* in preparing the soil flexibility matrix.

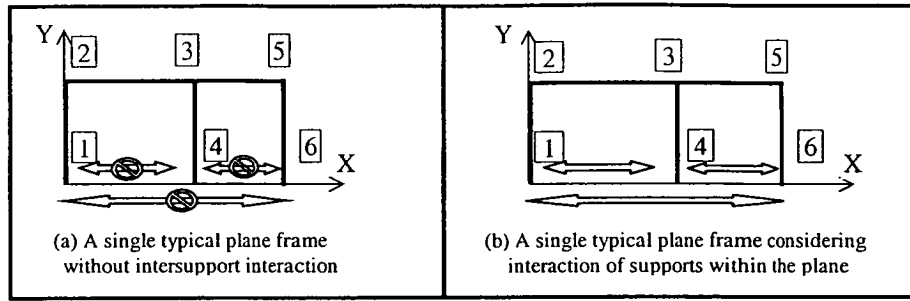


Figure 5.1. Illustration of a single plane frame considered for two cases: (a) No interactive support, (b) With interactive support

- c) Application of a series of typical single plane frames in parallel (in a direction perpendicular to the plane frame and on both sides of it, behind and in front) to form a space frame structure (Figure 5.2). These frames are under the same load as (or a portion of) the single plane frame. In Tables 5.1a through 5.1d, this option is marked by letter “S” as the last character of the file name which refers to *spatial frame* in the soil flexibility matrix.

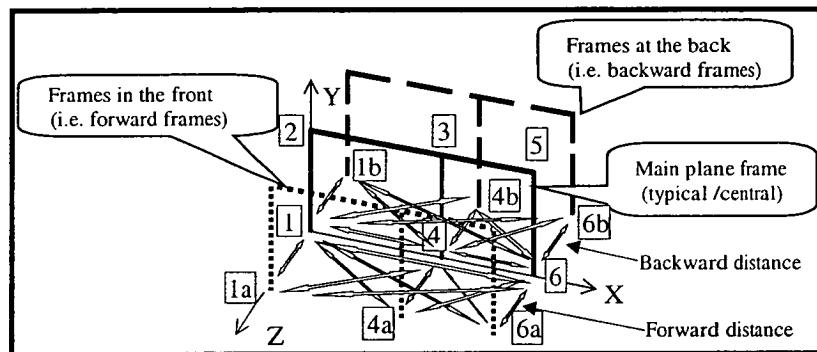


Figure 5.2 Illustration of a typical (main) plane frame and neighbouring parallel frames

The frame is analysed as being on an infinitely rigid foundation or on an elastic homogenous half-space (Figure 5.3) with isotropic / cross-anisotropic properties.

For simplicity of this example, the steel structural members are assumed prismatic (cross sections are uniform), and the material properties are constant and are:

- Cross-sectional area  $A = 0.0001 \text{ m}^2$
- Second moment of cross-sectional area about  $z$  axis  $I_z = 0.00010 \text{ m}^4$
- Young's modulus  $E = 200 \text{ GPa}$
- There are four frames adjacent to the original frame, two of them are located at rear (back frames) with distances of 4 and 6 meters to the next frame; and

the other two frames are at front (front frames) with distances of 4 and 4 meters to the next frame.

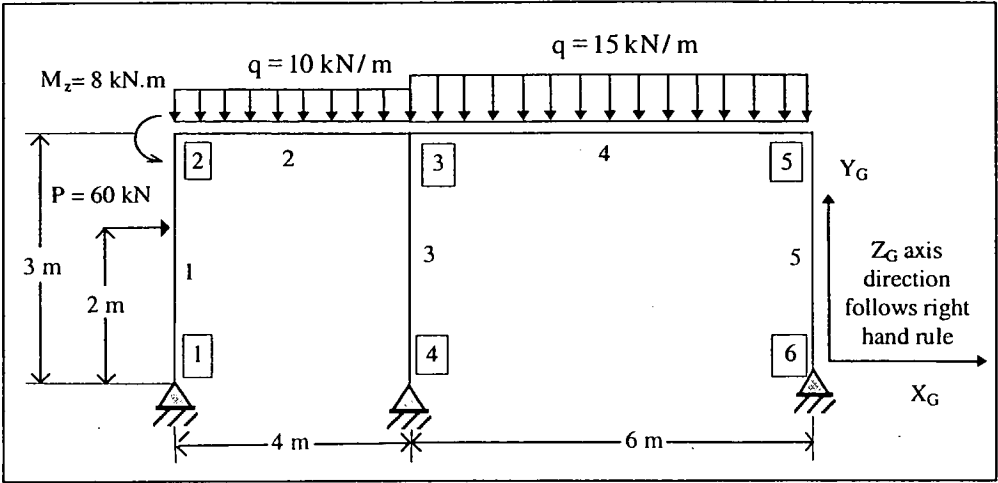


Figure 5.3. Profile of the structure and the external applied loads.

### 5.3. EXAMPLES STUDIED IN THIS PROJECT

As shown in the flow chart of Figure 5.4, four sets of examples (Table 5.1a to d) for isotropic and cross-anisotropic elastic half-space parameters are investigated.

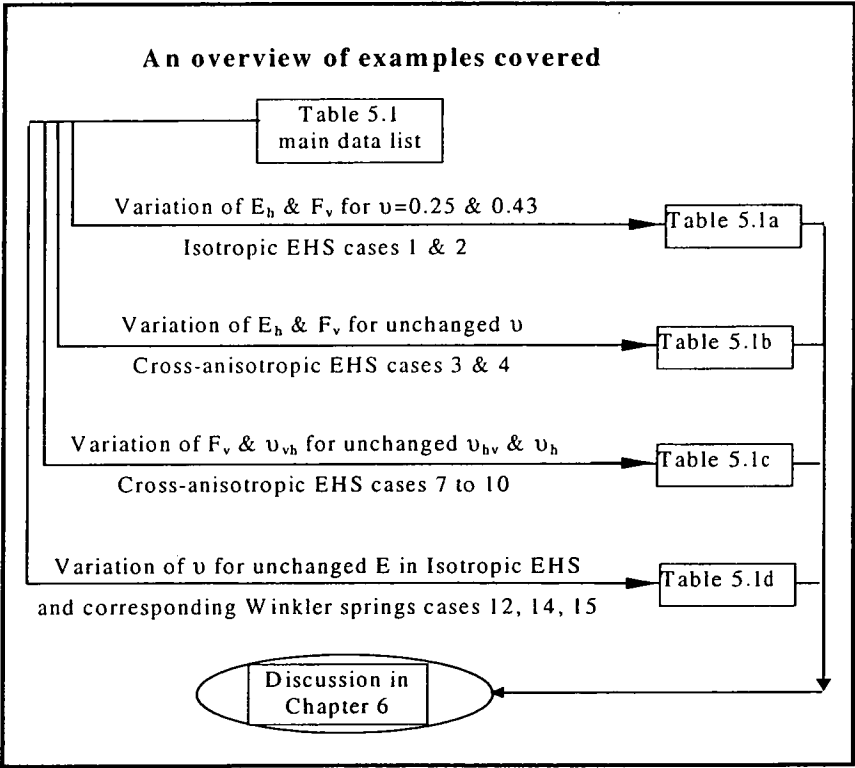


Figure 5.4 Flow chart for soil data in the examples

Values of moduli of elasticity (horizontal, vertical and shear moduli of elasticity  $E_H, E_V, F_V$ ) and Poisson's ratios ( $\nu_H, \nu_{HV}, \nu_{VH}$ ), (see section 2.4 for details) are given in Table 5.1. This data (soil data based on a typical Australian soils) is taken from studies conducted by CSIRO (Commonwealth Scientific and Industrial Research Organisation, Australia, Gerrard and Wardle 1973). In each set, one of the soil elastic parameters is varied and the response of the structure investigated.

Cases studied are listed in Table 5.1 and constitute four groups of parameters that are identified as follows:

- a) Cases 1 and 2 are related to isotropic elastic half-space with Poisson's ratios 0.25 and 0.43. The soil modulus of elasticity ( $E_H$ ) in horizontal direction is varied between 5 MPa and 150 MPa (soil shear modulus  $F_V$  is also proportionally varied), while the Poisson's ratio remains unchanged. These are used for input into Table 5.1a.
- b) Cases 3 and 5 are associated with cross-anisotropic half-space. The moduli of elasticity (horizontal, vertical and shear moduli of elasticity  $E_H, E_V, F_V$ ) are varied while the Poisson's ratios remain unchanged. These are used for input into Table 5.1.b.
- c) Cases 7 to 11 represent a cross-anisotropic half-space. Among the soil parameters, shear modulus of elasticity ( $F_V$ ) is varied to investigate the four distinct cases of cross-anisotropic half-space (refer to Chapter 2 for details of cross-anisotropic cases). Parameters in item 10 where  $\alpha=1$  and  $\beta=0$ , represent an isotropic property in the medium.

As mentioned in previous chapters, in a soil with isotropic properties there are three parameters involved: Young's modulus, Shear modulus and Poisson's ratio, ( $E, F$  and  $\nu$  respectively). Only two of these parameters are independent elastic constants and the following equation expresses the relationship for the third quantity.

$$F_V = \frac{E}{1 + \nu} \quad (5.1)$$

In a cross-anisotropic soil, six elastic parameters are defined:  $E_H$ ,  $E_V$ ,  $F_V$ ,  $\nu_H$ ,  $\nu_{HV}$  and  $\nu_{VH}$ . Only five of these quantities are independent and the following equation describes the inter-relationships for the sixth quantity

$$\frac{E_V}{E_H} = \frac{\nu_{VH}}{\nu_{HV}} \quad (5.2)$$

To define the equivalent for subgrade modulus  $k_s$ , it is assumed that the settlement of a circular rigid footing on Winkler medium should be the same as that for an identical footing resting on the elastic half-space (Hemsley 1987). Hence

$$k_s = \frac{2 E_s}{\pi R (1 - \nu_s^2)} \quad (5.3)$$

where  $R$  is the footing radius.

### 5.3.1. Nomenclature Applicable to Examples, Charts, Graphs and Tables

The following nomenclature is used in the tables and results:

- 1, 2 ,...:     Numbering of soil group (variation is only in one soil parameter)
- Iso:           Soil is "Isotropic" in the analysis
- Cis:           Soil is "Cross-anisotropic" in the analysis
- D:            Application of soil interaction is restricted to the individual supports of the single plane frame to obtain coefficients located in only the main *Diagonal blocks* of the soil flexibility matrix. Support nodes receive interaction only from the soil beneath them, i. e. *Isolated* interaction at nodes 1, 4 and 6 (see Figure 5.1a)
- P:            the coefficients in the soil flexibility matrix associated with the support interaction of the *Plane 2D* frame were considered. Support nodes 1, 4 and 6 receive *2D* interaction (see Figure 5.1b)
- S:            the coefficients in the soil flexibility matrix associated with all the supports of the plane frame as well as the neighbouring frames which



form a *Spatial 3D*- structure were considered (1, 4, 6, 1a, 4a, 6a, 1b, 4b, 6b, see Figure 5.2)

Figures 5.5 and 5.6 show the symbols to assigned to the bending moments and nodal displacements discussed in outputs, tables and graphs. From the outputs, the quantities associated with the bending moments and nodal displacements under investigation as shown in Figure 5.3 are defined as follows:

- $M_{21}$ : Bending moment in element 1 at end node 2
- $M_{32}$ : Bending moment in element 2 at end node 3
- $M_{34}$ : Bending moment in element 4 at end node 3
- $\Delta_{x1}, \Delta_{y1}$ : Displacements (translations) at node 1, in  $x$  and  $y$  directions
- $\Phi_{z1}$ : Angular displacement (rotation) at node 1, about  $z$  direction
- $\Delta_{x3}, \Delta_{y3}$ : Displacements (translations) at node 3, in  $x$  and  $y$  directions
- $\Phi_{z3}$ : Angular displacement (rotation) at node 3, about  $z$  direction

As shown in the examples, the frame dimensions and the elastic quantities associated with the structural material remain constant while soil elastic parameters are varied for study.

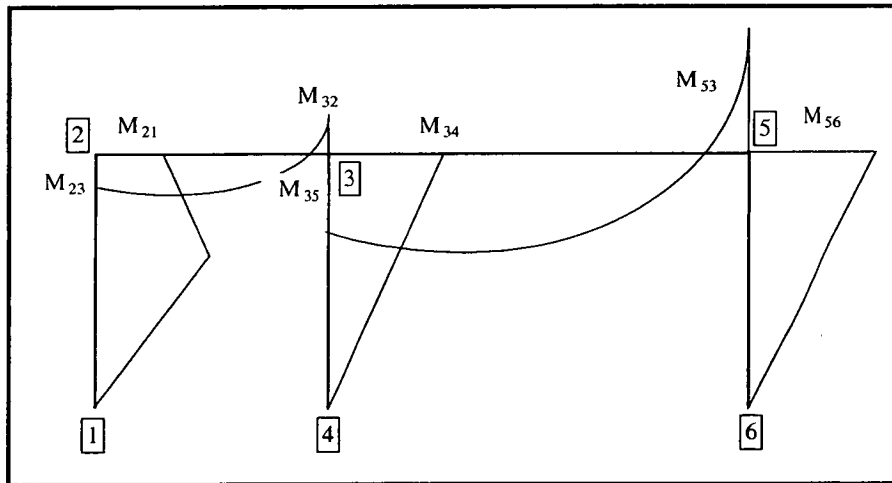


Figure 5.5 Nomination of symbols to element end-node moments associated with plane frame resulting from structural analyses

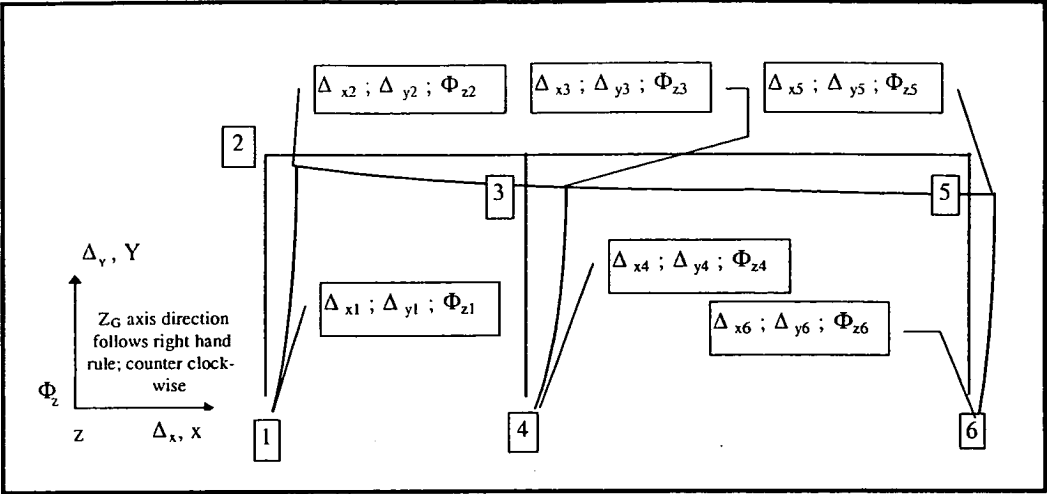


Figure 5.6 Nomination of symbols to nodal displacements associated with analysis

**Table of summary of soil properties**

sub-case	Soil type	soil constants					Reference and remarks
		$E_H/E_V$	$F_V/E_V$	$\nu_H$	$\nu_{HV}$	$\nu_{VH}$	
1	Isotropic	1.00	0.80	0.250	0.250	0.250	Gerrard, C. (1967) Fifth Aust-NZ Conf. On Soil Mechanics & Found. Engg. Soils:1, 2,..., 5
2		1.00	0.70	0.430	0.430	0.430	This sample represents a stiff isotropic pavement material overlaying a subgrade, Gerrard (1967)
3	Cross-anisotropic	1.50	0.90	0.250	0.300	0.200	This sample represents a typical clay with a high degree of saturation, Gerrard, C. (1967)
5		2.00	0.90	0.250	0.350	0.175	This sample represents a lightly overconsolidated clay, Gerrard, C. (1967)

		$E_H$ (MPa)	$E_V$ (MPa)	$\nu_{HV}$	$\nu_H$	$\nu_{VH}$	
7	Cross-anisotropic	30	21.450	0.400	0.400	0.466	Parameters of cross-anisotropic types are varied to meet three different cases for beta
8		30	23.000	0.400	0.400	0.466	
9		30	22.627	0.400	0.400	0.466	
10	Cross-anisotropic*	30	21.430	0.400	0.400	0.400	* Special case when cross-anisotropy is simplified to isotropy
11	Cross-anisotropic	3	9	4.620	0.300	0.300	Wardle, L.J. (1977)

		$E_H$ (MPa)	$k_s$ (MPa/m)	$\nu_{HV}$	$\nu_H$	$\nu_{VH}$
12	Isotropic	30	45.496	0.400	0.400	0.400
14	Isotropic	30	39.096	0.150	0.150	0.150
15	Isotropic	30	50.292	0.490	0.490	0.490

The above soil constants are the result of work carried out in the Department of Civil Engineering at the University of Melbourne under the direction of Professor A.J. Francis, which forms part of an investigation into the behaviour of road pavements which was sponsored by the Australian Road Research Board.

Soil constant definitions are as follows:

$E_H$  &  $E_V$  : Moduli of elasticity in horizontal and vertical directions;

$F_V$  : Shear modulus in vertical direction;

$\nu_H$  : Poisson's ratio in horizontal direction;

$\nu_{VH}$  : Poisson's ratio for the effect of vertical stress on horizontal strain;

$\nu_{HV}$  : Poisson's ratio for the effect of horizontal stress on vertical strain;

where  $E_H * \nu_{VH} = E_V * \nu_{HV}$  and  $F_H (1+\nu_H) = E_H$

Details of isotropic elastic half-space soil parameters corresponding to sub-cases 1-20 (Gerrard 1967)								
Sub-case	File name	Sub-case	Soil model	Horizontal and vertical elasticity moduli (MPa)	Shear modulus (MPa)	Poisson's ratios	$\alpha^2$	$\beta^2$
				$E_H = E_V$	$F_v = E/(1+\nu)$	$\nu_{HV} = \nu_H = \nu_{VH}$		
1	1-IsoD,P,S	Case 1 from Table 5.1	Isotropic EHS (variable $E_H$ )	5	4	0.250	N/A	N/A
2	2-IsoD,P,S			10	8	0.250	N/A	N/A
3	3-IsoD,P,S			15	12	0.250	N/A	N/A
4	4-IsoD,P,S			20	16	0.250	N/A	N/A
5	5-IsoD,P,S			30	24	0.250	N/A	N/A
6	6-IsoD,P,S			40	32	0.250	N/A	N/A
7	7-IsoD,P,S			50	40	0.250	N/A	N/A
8	8-IsoD,P,S			70	56	0.250	N/A	N/A
9	9-IsoD,P,S			120	96	0.250	N/A	N/A
10	10-IsoD,P,S			150	120	0.250	N/A	N/A
11	11-IsoD,P,S	Case 2 from Table 5.1	Isotropic EHS (variable $E_H$ )	5	3	0.430	N/A	N/A
12	12-IsoD,P,S			10	7	0.430	N/A	N/A
13	13-IsoD,P,S			15	10	0.430	N/A	N/A
14	14-IsoD,P,S			20	14	0.430	N/A	N/A
15	15-IsoD,P,S			30	21	0.430	N/A	N/A
16	16-IsoD,P,S			40	28	0.430	N/A	N/A
17	17-IsoD,P,S			50	35	0.430	N/A	N/A
18	18-IsoD,P,S			70	49	0.430	N/A	N/A
19	19-IsoD,P,S			120	84	0.430	N/A	N/A
20	20-IsoD,P,S			150	105	0.430	N/A	N/A

Details of cross-anisotropic soil parameters corresponding to sub-cases 1- 10 and 21-30 (Gerrard 1967)											
Sub-case	File name	Sub-case	Soil model	Horizontal elasticity modulus (MPa)	Vertical elasticity modulus (MPa)	Shear modulus (MPa)	Poisson's ratios			$\alpha^2$	$\beta^2$
				$E_H$	$E_V$	$F_V$	$\nu_{HV}$	$\nu_H$	$\nu_{VH}$		
1	1-CisoD,P,S	Case 3 from Table 5.1	Cross-anisotropic EHS	39	26	23.4	0.300	0.250	0.200	$\alpha^2 < 1$	$\beta^2 = 0.0075 > 0$
2	2-CisoD,P,S			40.5	27	24.3	0.300	0.250	0.200		
3	3-CisoD,P,S			42	28	25.2	0.300	0.250	0.200		
4	4-CisoD,P,S			43.5	29	26.1	0.300	0.250	0.200		
5	5-CisoD,P,S			45	30	27	0.300	0.250	0.200		
6	6-CisoD,P,S			46.5	31	27.9	0.300	0.250	0.200		
7	7-CisoD,P,S			48	32	28.8	0.300	0.250	0.200		
8	8-CisoD,P,S			49.5	33	29.7	0.300	0.250	0.200		
9	9-CisoD,P,S			51	34	30.6	0.300	0.250	0.200		
10	10-CisoD,P,S			52.5	35	31.5	0.300	0.250	0.200		
21	21-CisoD,P,S	Case 5 from Table 5.1	Cross-anisotropic EHS	150	75	67.5	0.350	0.250	0.175	$\alpha^2 < 1$	$\beta^2 = 0.126 > 0$
22	22-CisoD,P,S			152	76	68.4	0.350	0.250	0.175		
23	23-CisoD,P,S			154	77	69.3	0.350	0.250	0.175		
24	24-CisoD,P,S			156	78	70.2	0.350	0.250	0.175		
25	25-CisoD,P,S			158	79	71.1	0.350	0.250	0.175		
26	26-CisoD,P,S			160	80	72	0.350	0.250	0.175		
27	27-CisoD,P,S			162	81	72.9	0.350	0.250	0.175		
28	28-CisoD,P,S			164	82	73.8	0.350	0.250	0.175		
29	29-CisoD,P,S			166	83	74.7	0.350	0.250	0.175		
30	30-CisoD,P,S			168	84	75.6	0.350	0.250	0.175		

Details of soil parameters corresponding to sub-case 32- 36

Sub-case	File name	Sub-case	Soil model	Horizontal & vertical elasticity moduli (MPa)		Vertical shear modulus (MPa)	Poisson's ratios			$\alpha^2$	$\beta^2$
				$E_H$	$E_v$	$F_v$	$v_{HV}$	$v_H$	$v_{VH}$		
35	35-CisoD,P,S	10	Special cross-aniso	30	$=E_H \cdot V_{VH}/V_{HV}=30$	21.429	0.400	0.400	0.400	1.000	zero
32	32-CisoD,P,S	7	Cross-anisotropic	30	$=E_H \cdot V_{VH}/V_{HV}=34.950$	21.450	0.400	0.400	0.466	<>1	positive
34	34-CisoD,P,S	9	Cross-anisotropic	30	$=E_H \cdot V_{VH}/V_{HV}=34.952$	22.627	0.400	0.400	0.466	<>1	zero
33	33-CisoD,P,S	8	Cross-anisotropic	30	$=E_H \cdot V_{VH}/V_{HV}=34.951$	23.000	0.400	0.400	0.466	<>1	negative
36	36-CisoD,P,S	11	Isotropic	30	$=E_H \cdot V_{VH}/V_{HV}=30$	$=E/(1+\nu)=21.429$	0.400	0.400	0.400	N/A	N/A

Table 5.1c Listing of elastic properties for isotropic and Cross-anisotropic soil sub-cases 32-36 (Wardle 1977)

Sub-case	File Name	Sub-case	Soil Model	Horizontal & vertical elasticity moduli (MPa)		Corresponding Winkler spring#	Poisson's ratios		
				$E_H$	$E_v$	$k_s$ (MPa/m)	$v_{HV}$	$v_H$	$v_{VH}$
37	37-IsoD, P, S	12	Isotropic	30	30	49.495	0.400	0.400	0.400
39	WIso-1D, P, S	14	Isotropic	30	30	39.096	0.150	0.150	0.150
40	WIso-2D, P, S	15	Isotropic	30	30	50.292	0.490	0.490	0.490

Table 5.1d Listing of elastic properties for isotropic soil sub-cases 37, 39, 40 and its equivalent of Winkler spring

#: The Winkler spring coefficient corresponding to an isotropic soil (see equation 5.3 ) for a circle with  $D=1$  m

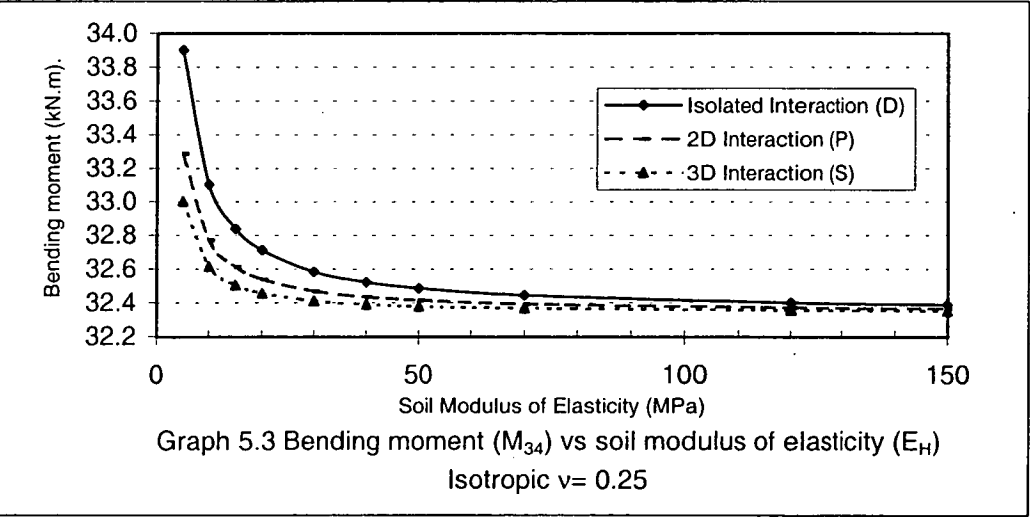
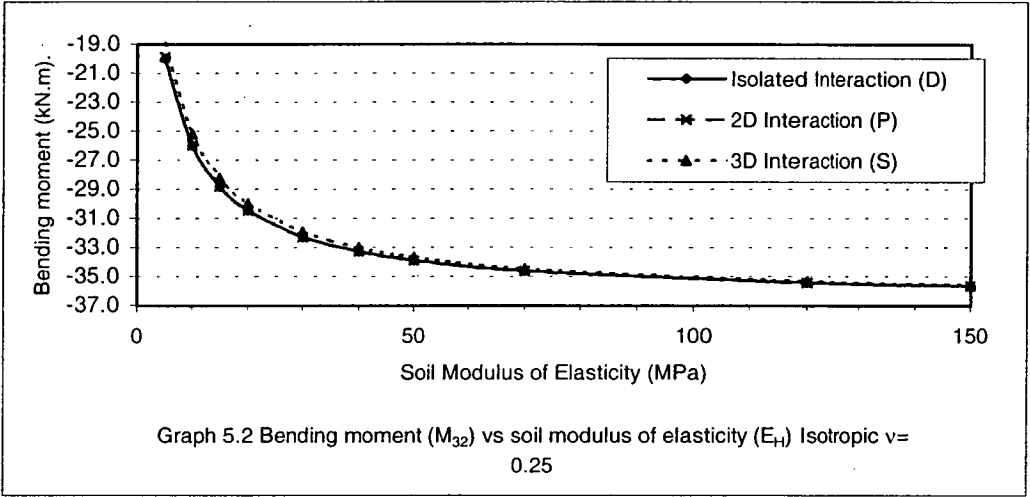
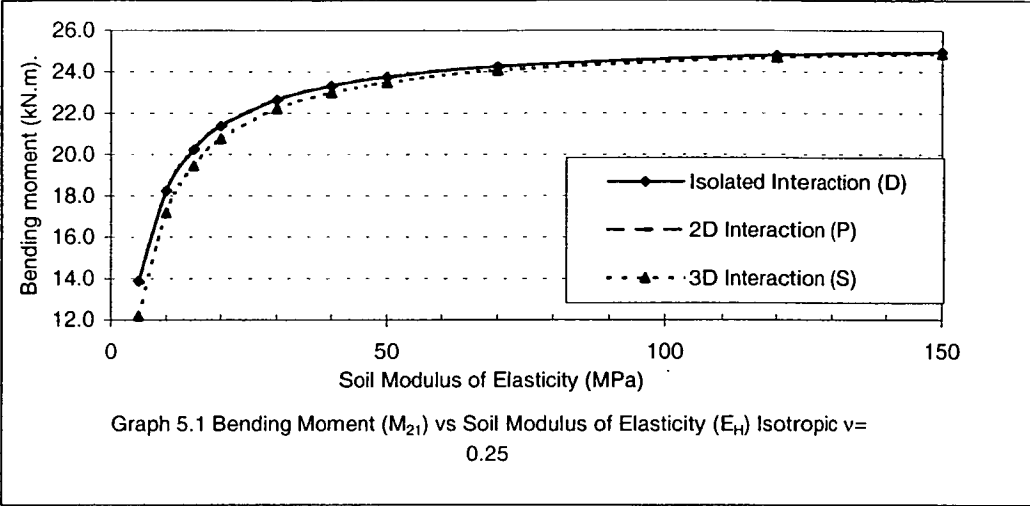
### 5.3.2. Examples of Analysis of Frame Founded on Isotropic Soil

Two main types of soil with isotropic properties are considered in this subsection. An example of an output graph and the associated data is presented below. In the example below, soil properties corresponding to items 1 to 10 in Table 5.1a which relates to case 1 in Table 5.1 (i. e. Isotropic elastic half-space  $E_H=5$  to  $150 \text{ MPa}$ ,  $\nu=0.25$ ). Element bending moment  $M_{21}$  from the *SASIAP* analysis is plotted against soil modulus of elasticity for three cases of support interaction: isolated interaction ( $D$ ), interaction within plane (i.e. 2D interaction  $P$ ), and interaction from spatial combination (i.e. 3D interaction  $S$ ). The analysis results for element end-moments ( $M_{21}$ ,  $M_{33}$ ,  $M_{34}$ ) and nodal displacements ( $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$ ,  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$ ) are presented in Graphs 5.1 - 5.9 and Graphs 5.10 - 5.18 for case 1 and case 2 respectively (Table 5.1). The numerical values associated with these graphs are also available in the Appendix A5.1.

<b>Bending Moment (<math>M_{21}</math>) against Soil Horizontal Modulus of Elasticity Isotropic (<math>E_H</math>)</b> I. Isotropic Soil (See Table 5.1) $\nu = 0.25$				
item	E soil (MPa)	D group (kN-m)	P group (kN-m)	S group (kN-m)
1-Iso	5	13.8708	13.8171	12.1923
2-Iso	10	18.2303	18.2473	17.1872
3-Iso	15	20.2241	20.2531	19.4632
4-Iso	20	21.3693	21.3996	20.7698
5-Iso	30	22.6349	22.6619	22.2136
6-Iso	40	23.3191	23.3423	22.9942
7-Iso	50	23.7478	23.7679	23.4834
8-Iso	70	24.2556	24.2712	24.0629
9-Iso	120	24.8058	24.8158	24.6910
10-Iso	150	24.9640	24.9722	24.8716

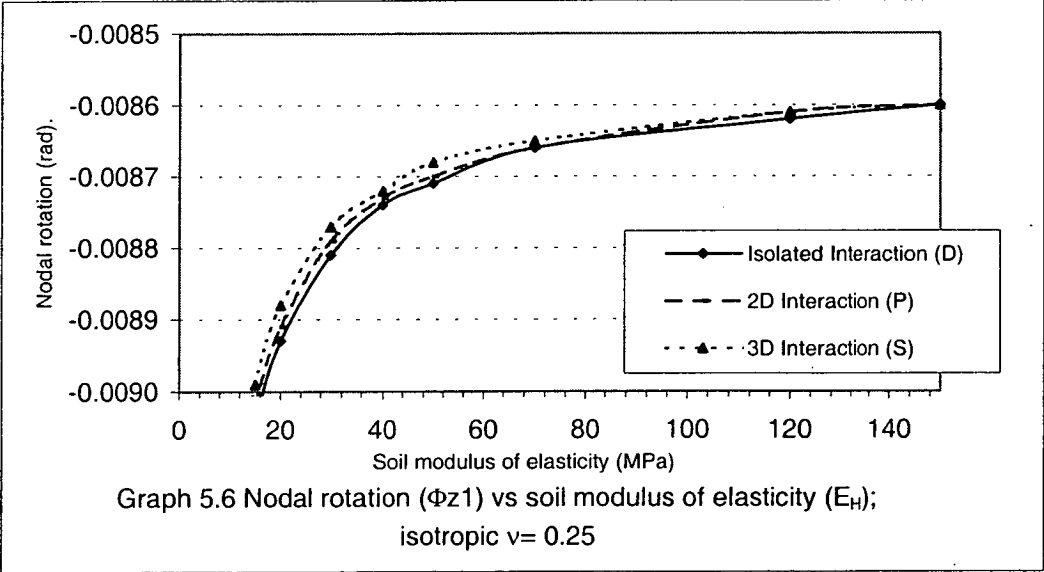
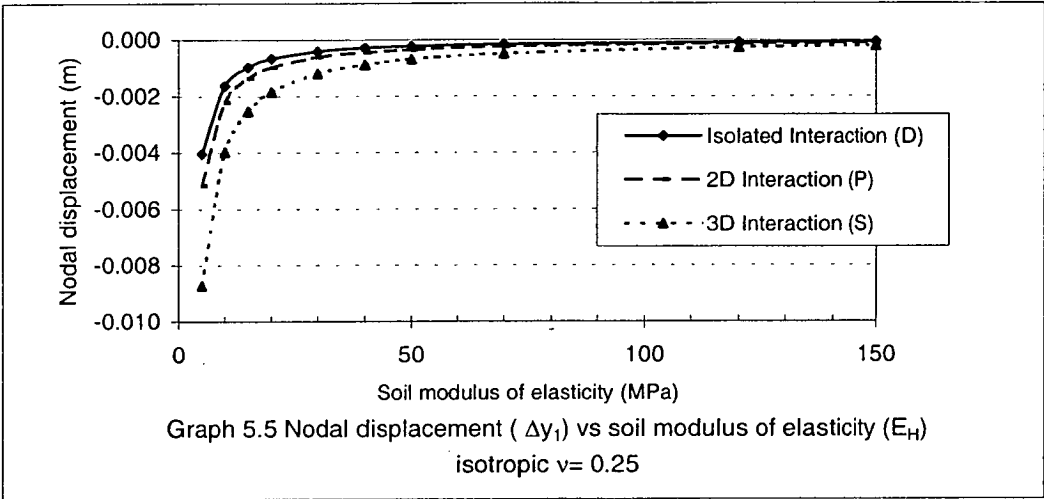
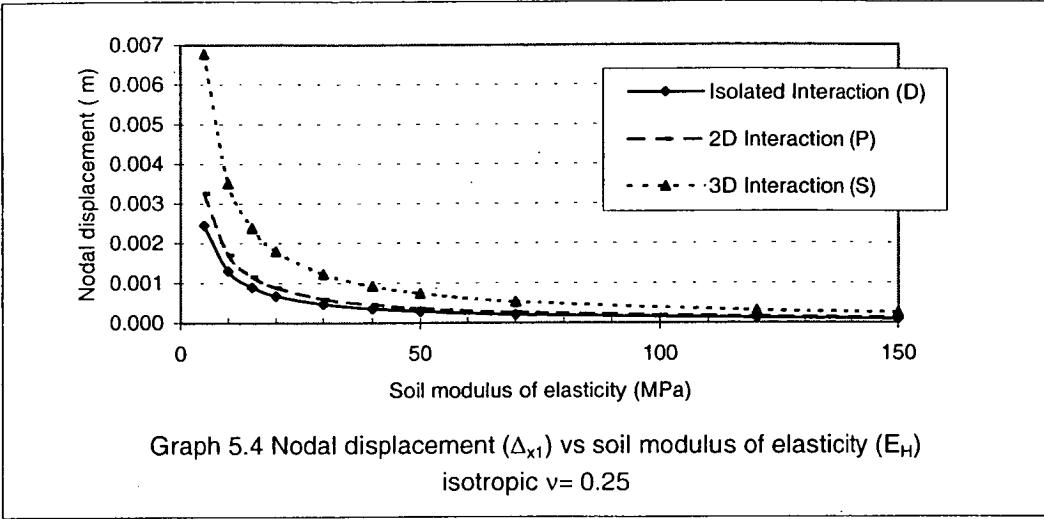
Table 5.2 Output data for element bending moment  $M_{21}$  from structure analysis when soil modulus of elasticity (Young's modulus) varies between 5 and  $150 \text{ MPa}$  for three different interaction cases.

Data associated with Graphs 5.1 to 5.18 is available in Appendix A5.1. Table 5.2 is presented as an example of values that is used for Graph 5.1. Tables associated with Graphs 5.2 to 5.18 are available in Appendix A5 for reference.

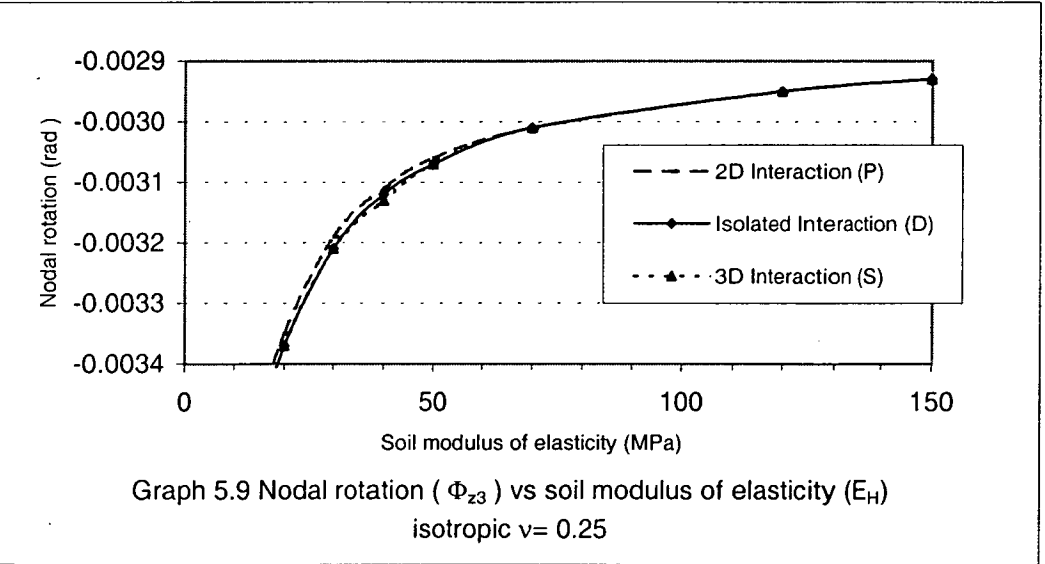
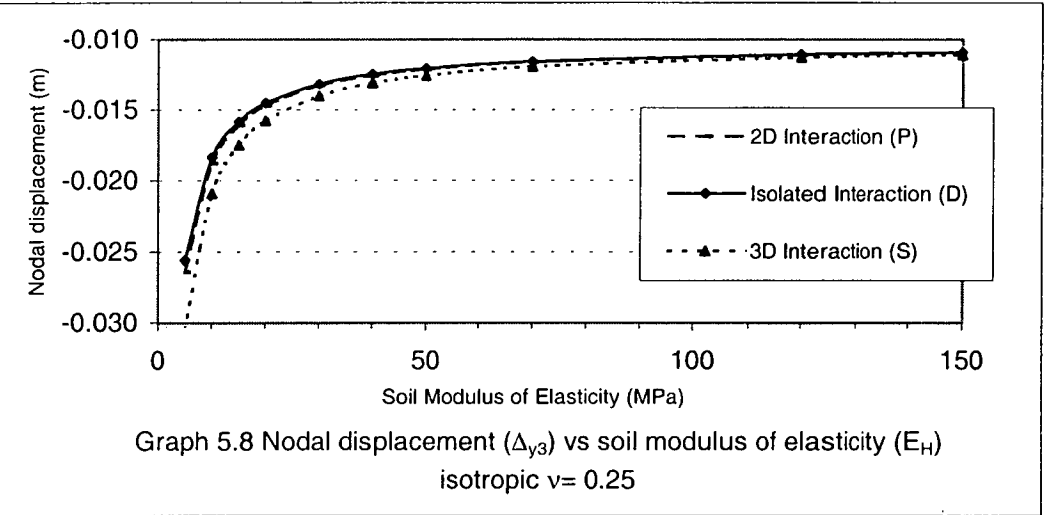
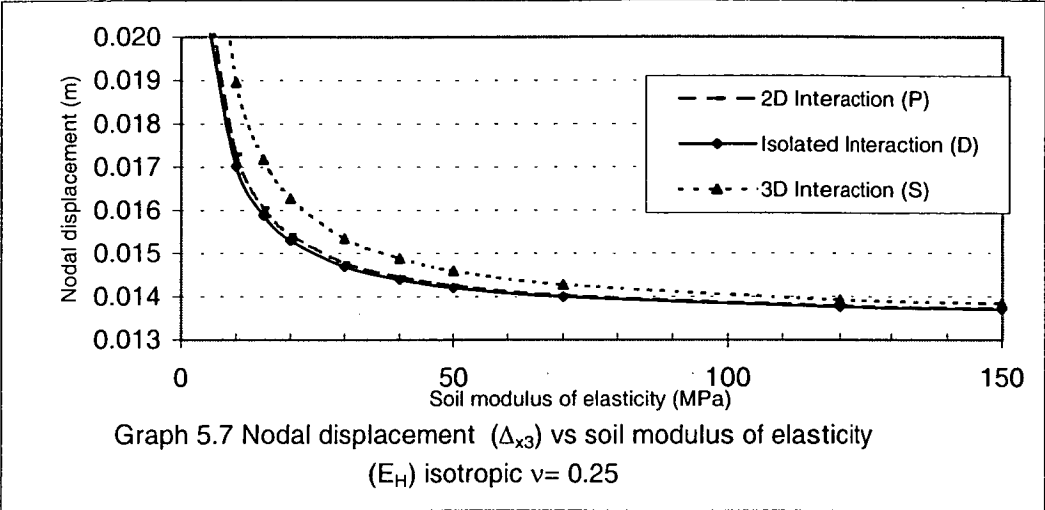


Graphs 5.1, 5.2 and 5.3 show the bending moment in the members 2-1, 2-3 and 3-4 ( $M_{21}$ ,  $M_{32}$  and  $M_{34}$ ) respectively for isotropic soil with  $\nu=0.25$  (See Table 5.1a)

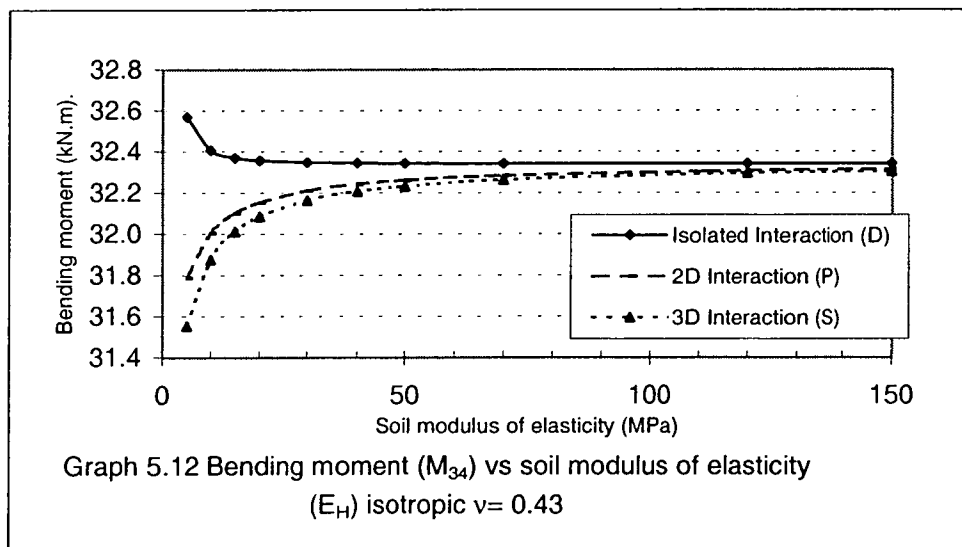
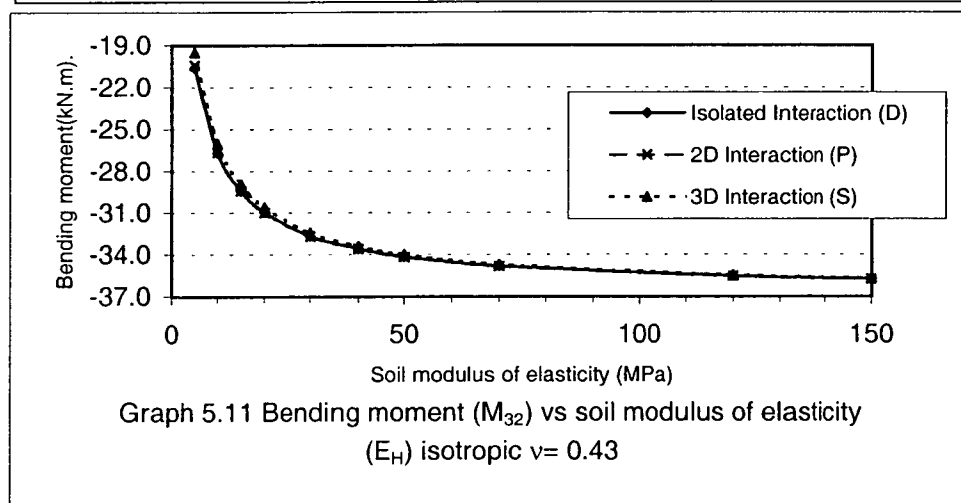
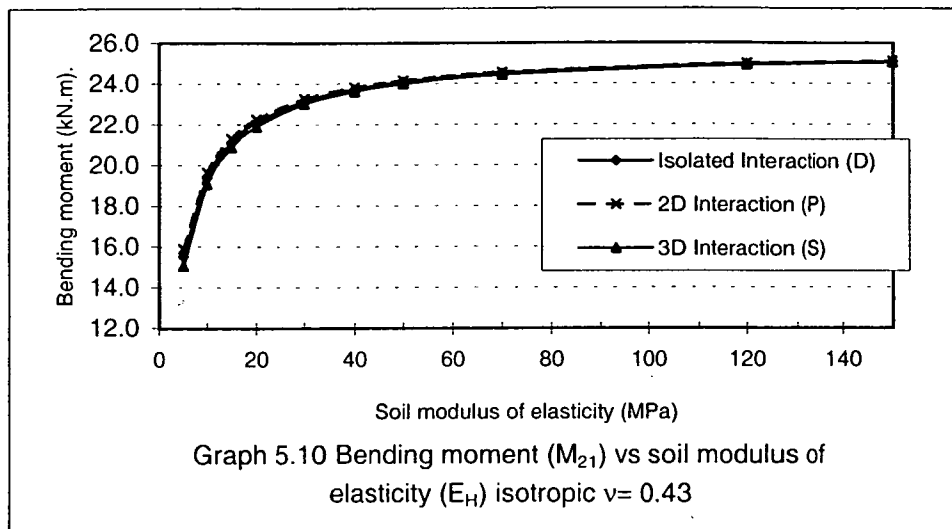




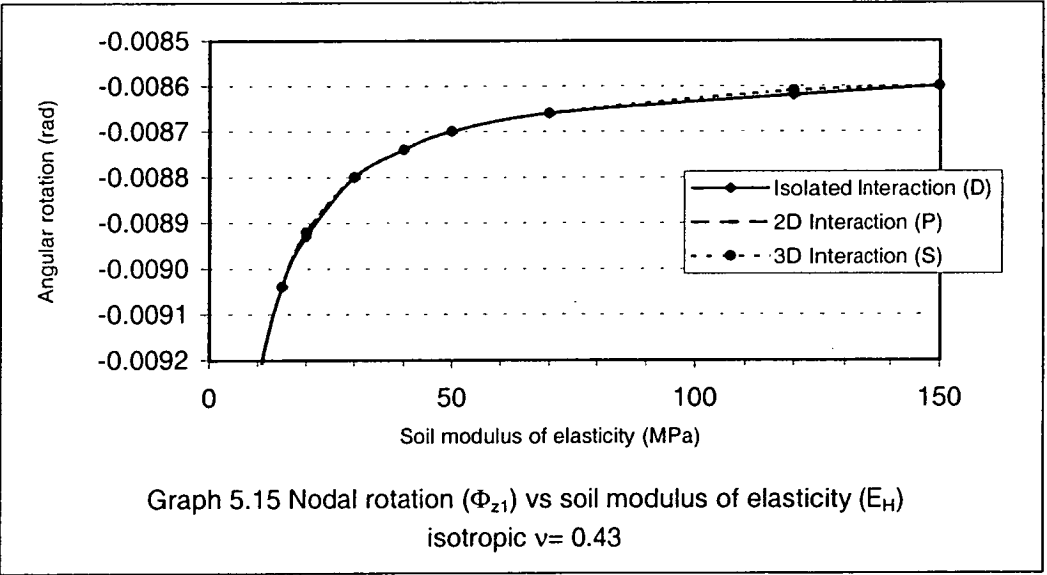
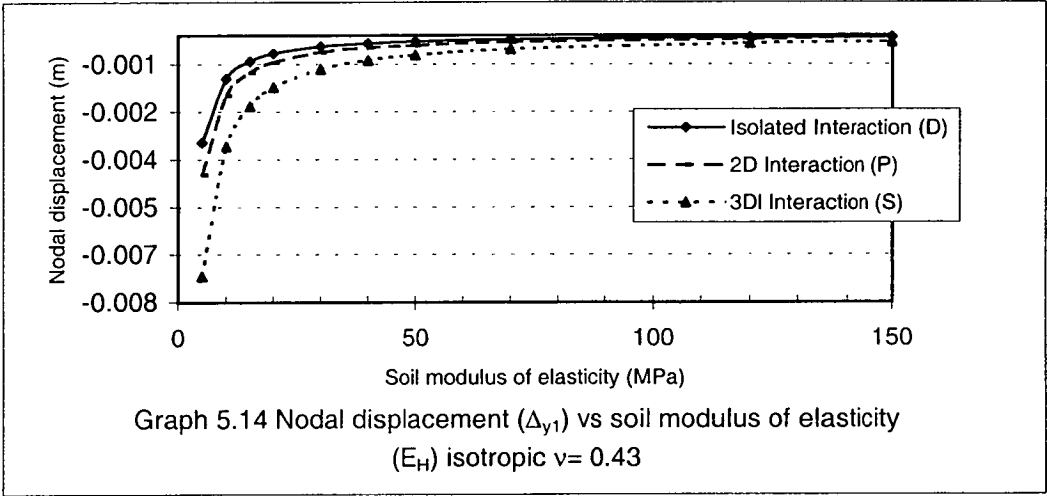
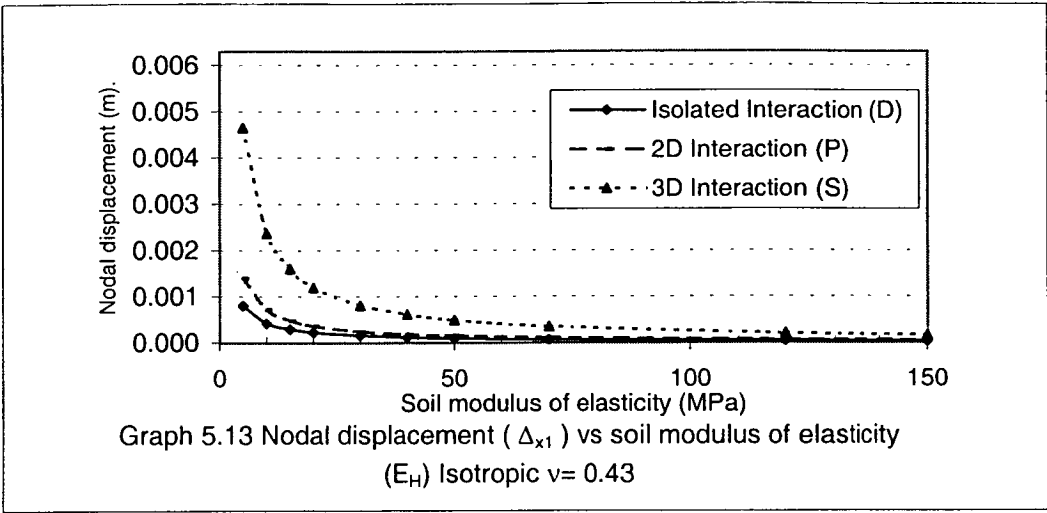
Graphs 5.4, 5.5 and 5.6 show the nodal displacements  $\Delta_{x1}, \Delta_{y1}$  and nodal rotation  $\Phi_{z1}$  respectively at node 1 for isotropic soil with  $\nu = 0.25$  (see Table 5.1a)



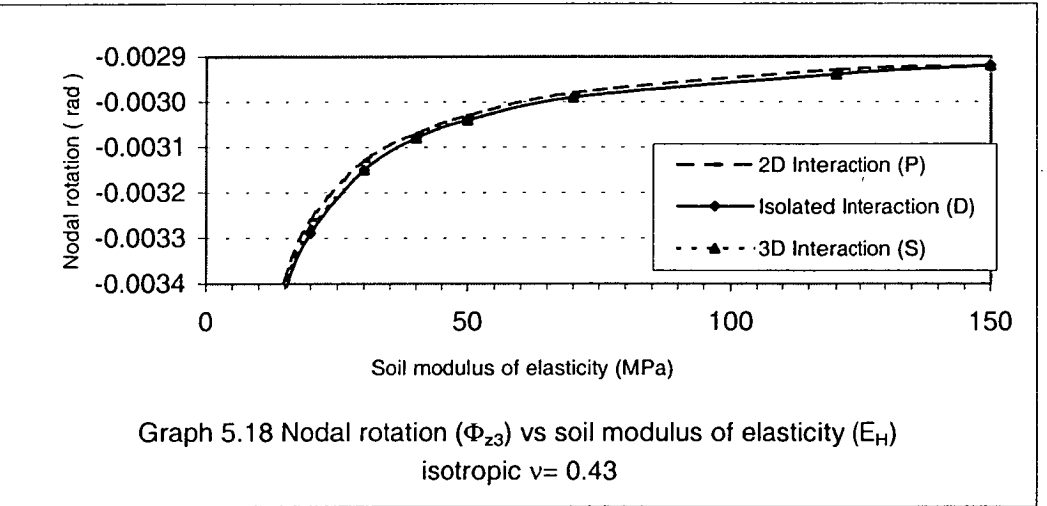
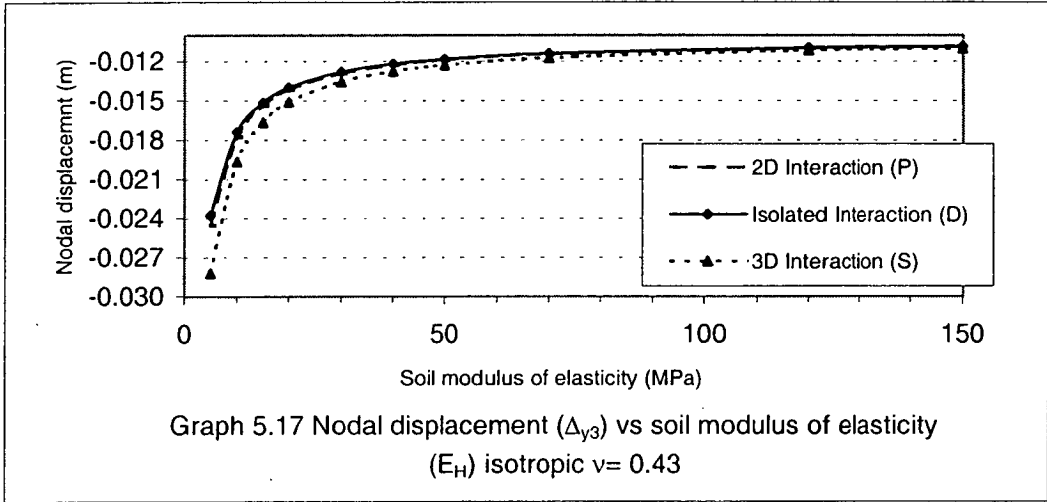
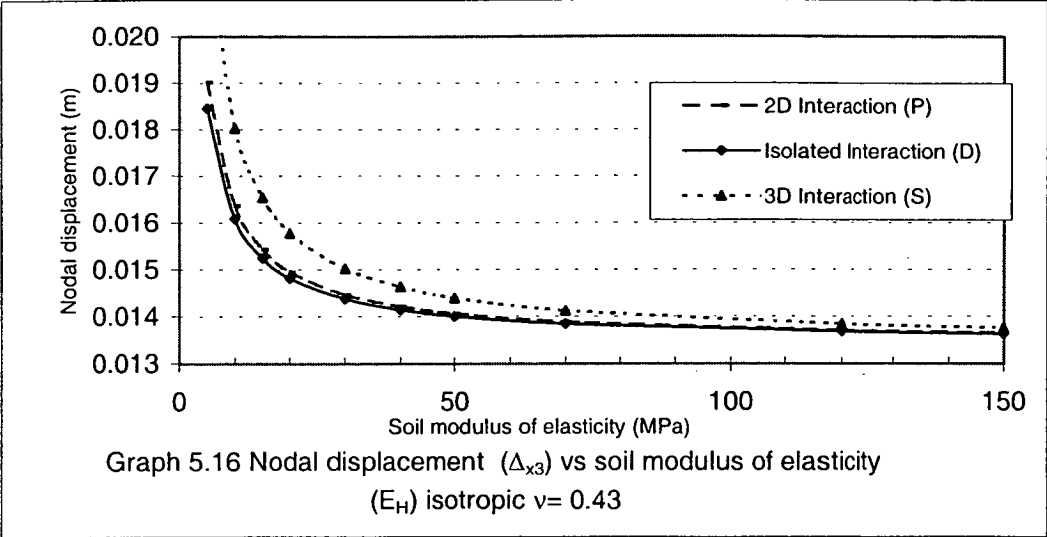
Graphs 5.7, 5.8 and 5.9 show nodal displacements  $\Delta_{x3}, \Delta_{y3}$  and nodal rotation  $\Phi_{z3}$  respectively at node 3 for isotropic soil  $\nu = 0.25$  (see Table 5.1a)



Graphs 5.10, 5.11 and 5.12 show bending moments ( $M_{21}$ ,  $M_{32}$ ,  $M_{34}$ ) in members 1-2, 2-3 and 3-4 respectively for  $\nu=0.43$  in isotropic soil (see Table 5.1a)



Graphs 5.13, 5.14 and 5.15 show nodal displacements  $\Delta_{x1}, \Delta_{y1}$  and nodal rotation  $\Phi_{z1}$  respectively at node 1 for isotropic soil  $\nu = 0.43$  (see Table 5.1a)



Graphs 5.16, 5.17 and 5.18 show nodal displacements  $\Delta_{x3}, \Delta_{y3}$  and nodal rotation  $\Phi_{z3}$  respectively at node 3 for isotropic soil  $\nu = 0.43$  (see Table 5.1a)

### 5.3.3. Set One Examples for Cross-anisotropic Soil

This section covers cases 3 and 5, shown in Table 5.1 and Table 5.1b. These cases use increments in vertical modulus of elasticity  $E_v$  as the reference parameter for two selected ranges of: 26 to 35 MPa and 75 to 84 MPa, using the following relations obtained by Gerrard, C. (1967) between elastic parameters. For the case 3:

$$\frac{E_H}{E_v} = 1.50 \text{ and } \frac{F}{E_v} = 0.90, \text{ and for the case 5: } \frac{E_H}{E_v} = 2.00 \text{ and } \frac{F}{E_v} = 0.90. \text{ In both}$$

cases, the remaining elastic quantities are unchanged.

For consistency in the structure dimension and eliminating the variation caused by load, the same loads used earlier apply and are presented in the figure below are applicable to the structure.

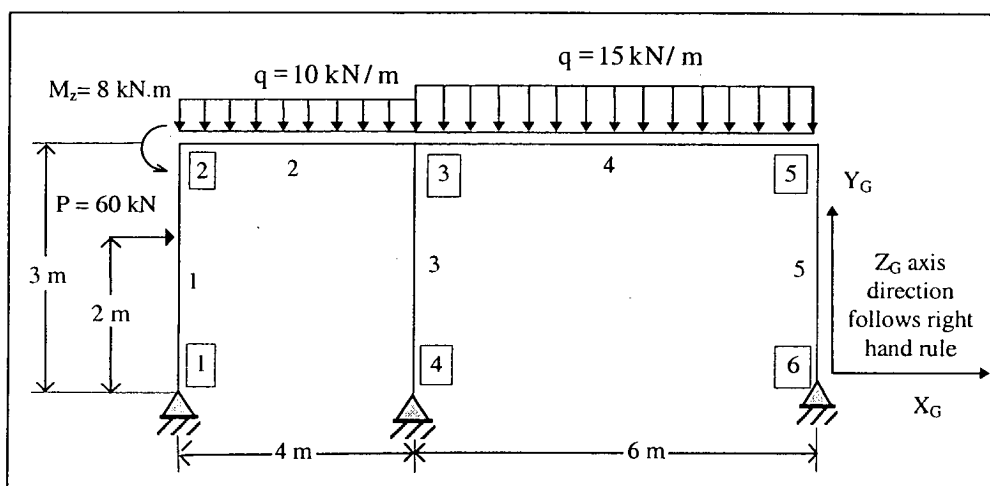
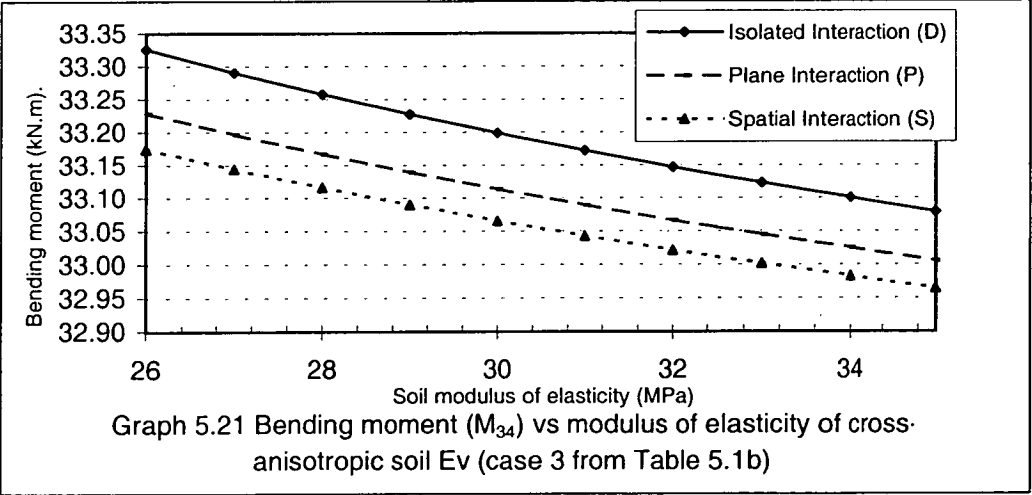
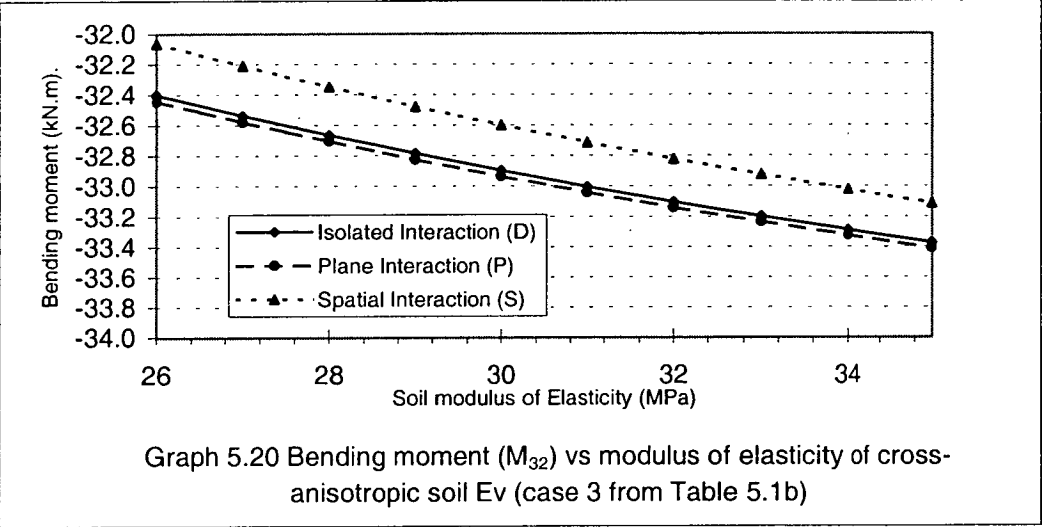
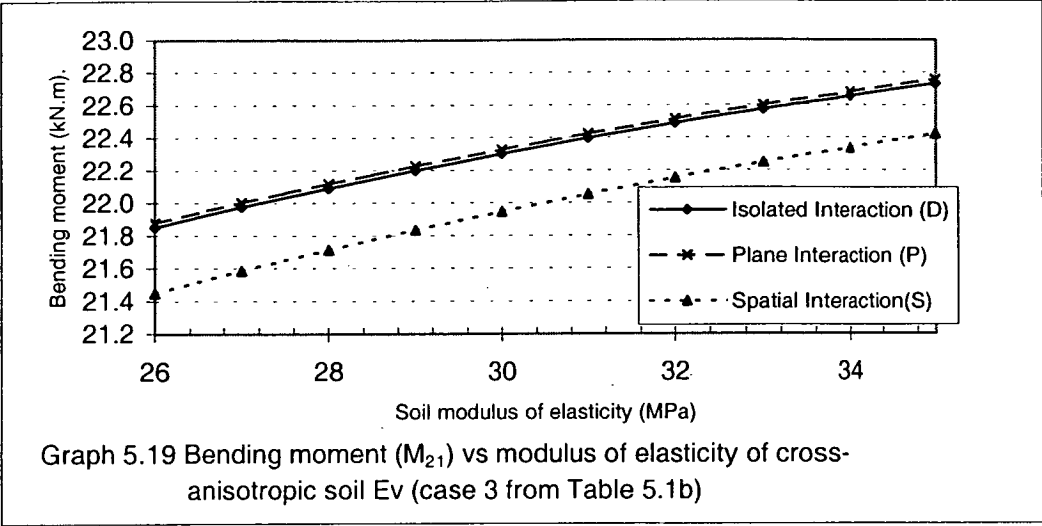
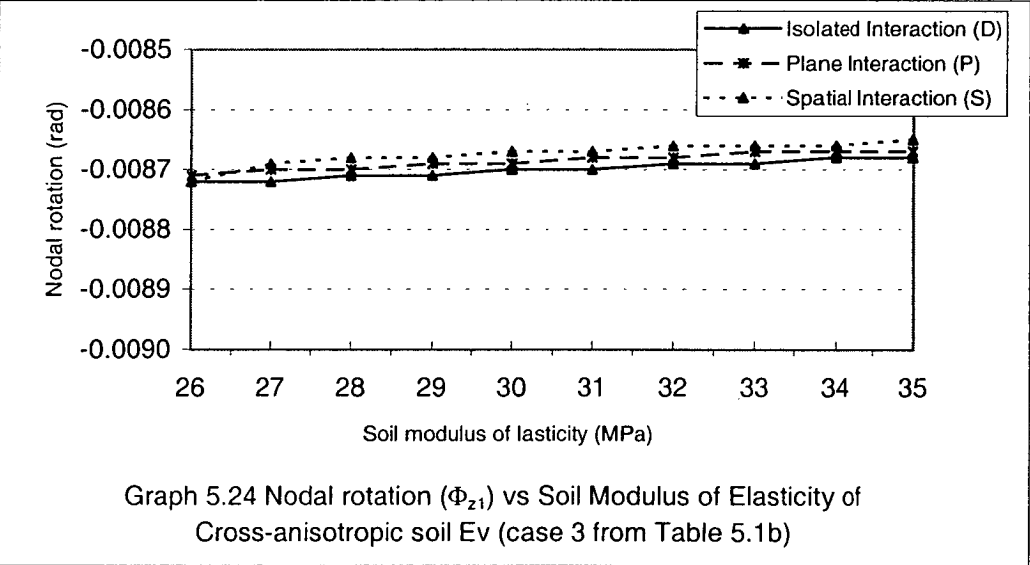
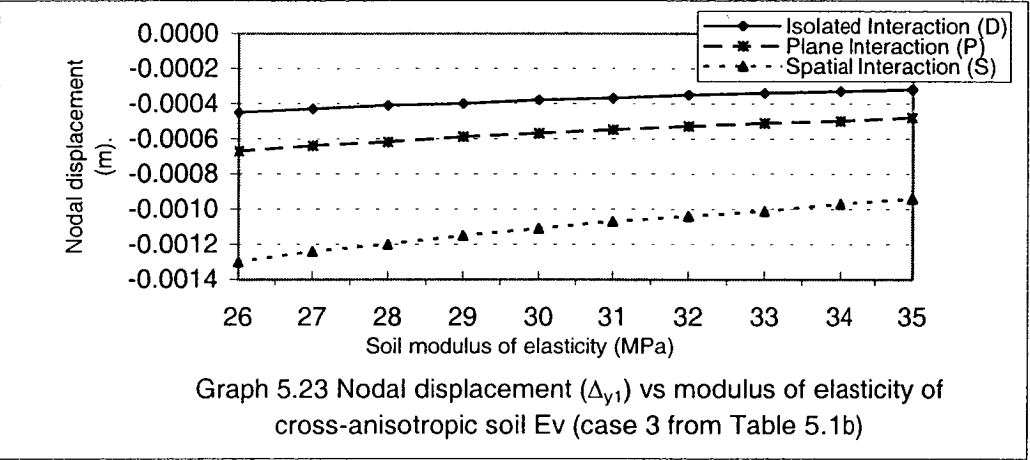
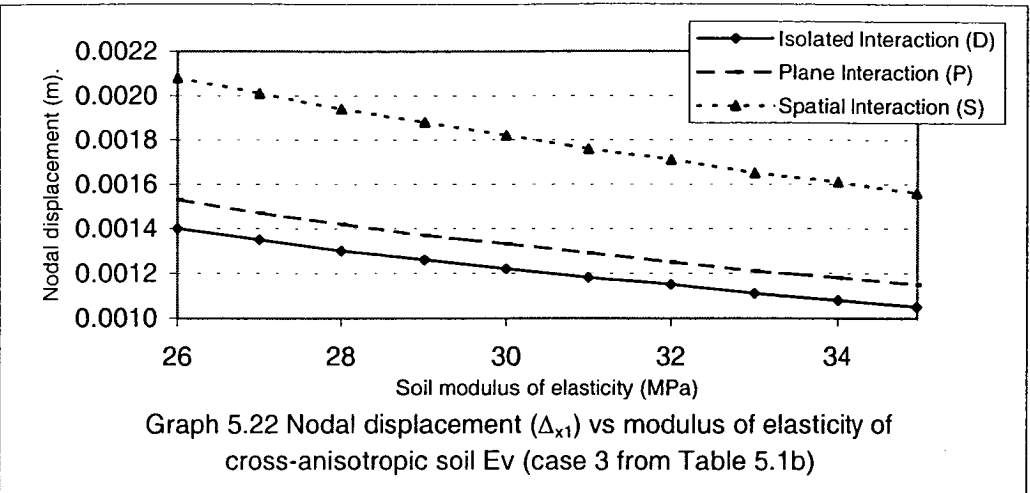


Figure 5.3. Profile of the structure and external applied loads.

Graphs 5.19-5.36 present analysis output for element end-moments ( $M_{21}$ ,  $M_{33}$ ,  $M_{34}$ ) and nodal displacements ( $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$ ,  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$ ) based on the elastic parameters from Tables 5.1b and 5.1c. Numerical values corresponding to these graphs are also available in Appendix A5-2.

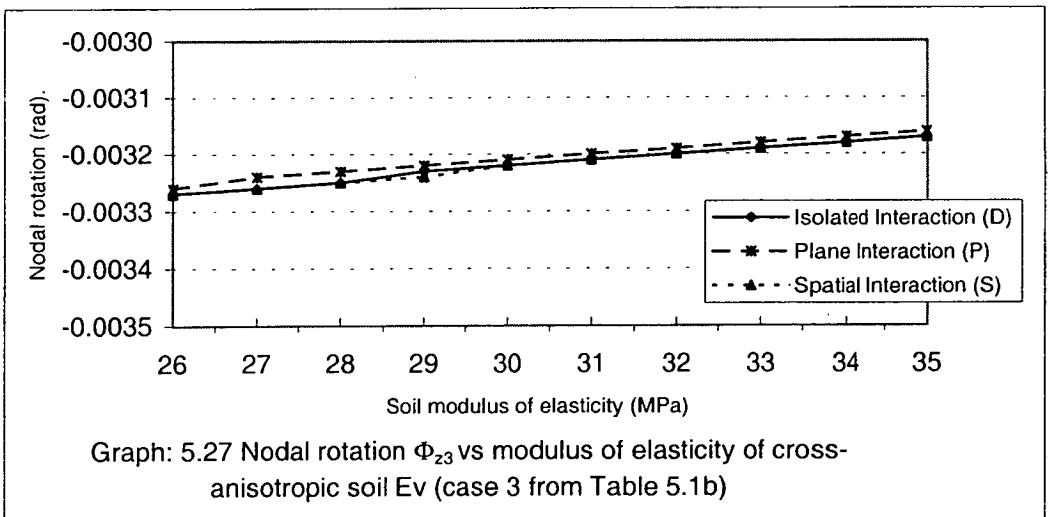
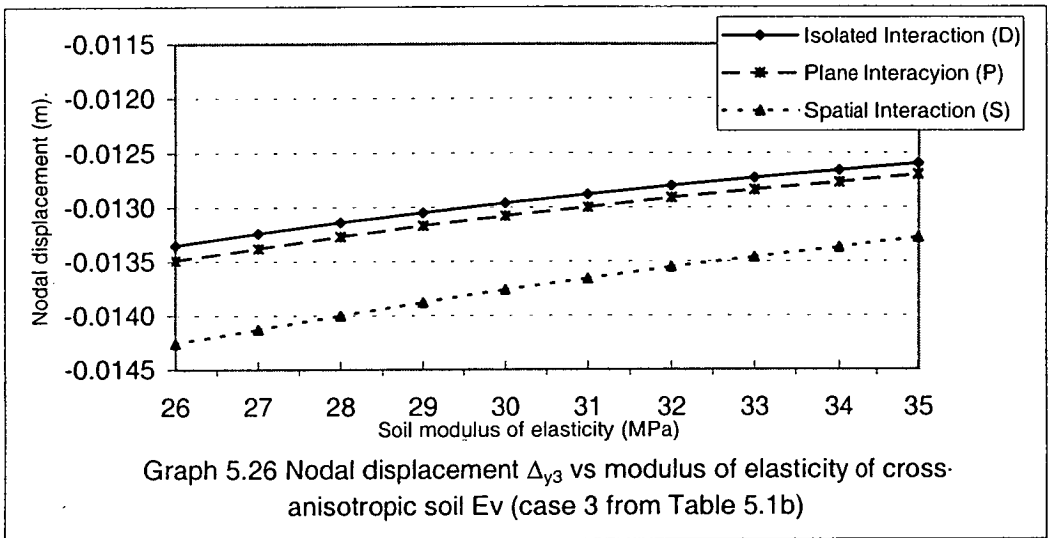
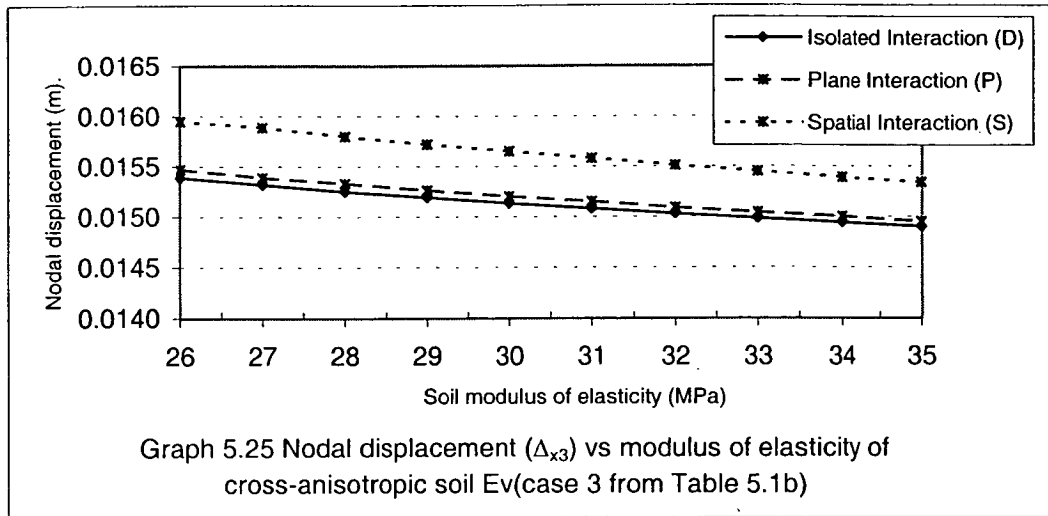


Graphs 5.19, 5.20 and 5.21 show variation bending moments ( $M_{21}$ ,  $M_{32}$ ,  $M_{34}$ ) in members 1-2, 2-3 and 3-4 respectively for  $E_v$  range between 26-35 MPa in cross-anisotropic soil (case 3 in Table 5.1b)

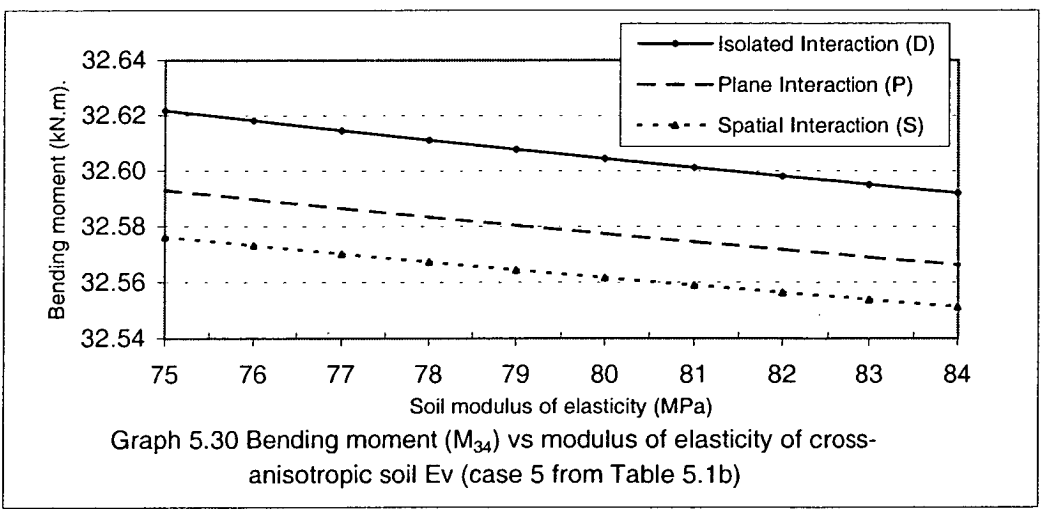
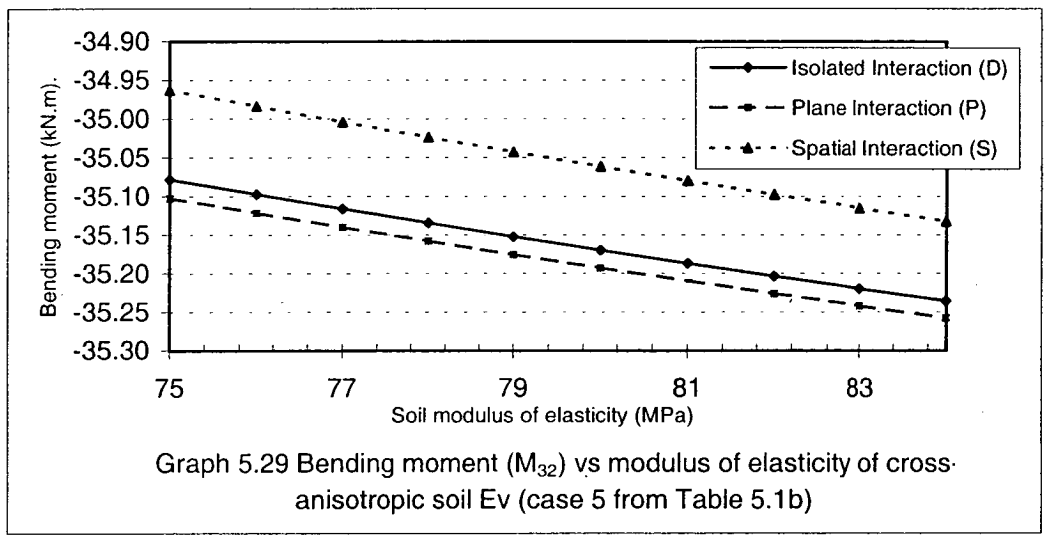
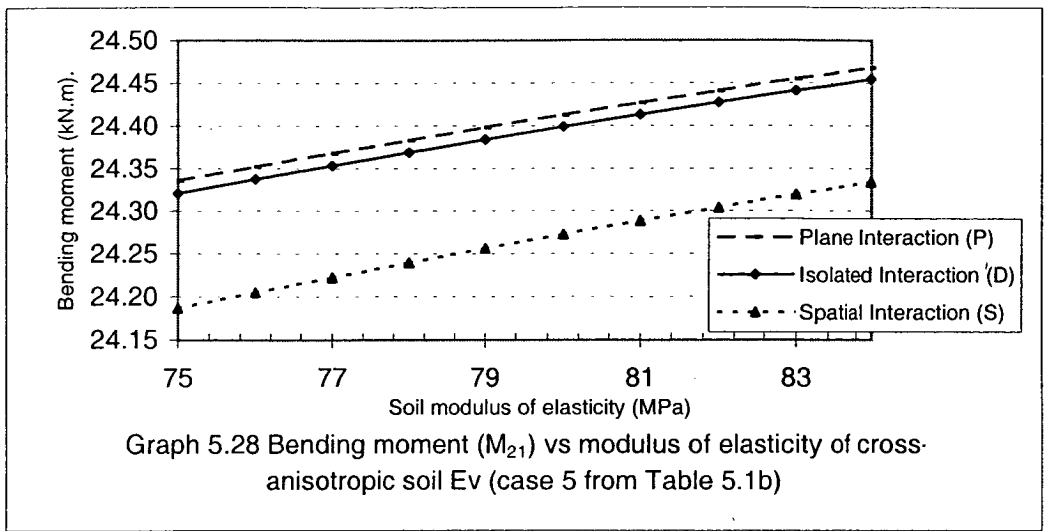


Graphs 5.22, 5.23 and 5.24 show variation of nodal displacements  $\Delta_{x1}, \Delta_{y1}$  and nodal rotation  $\Phi_{z1}$  respectively at node 1 for  $E_v$  range between 26-35 MPa in cross-anisotropic iso soil (case 3 in Table 5.1b)

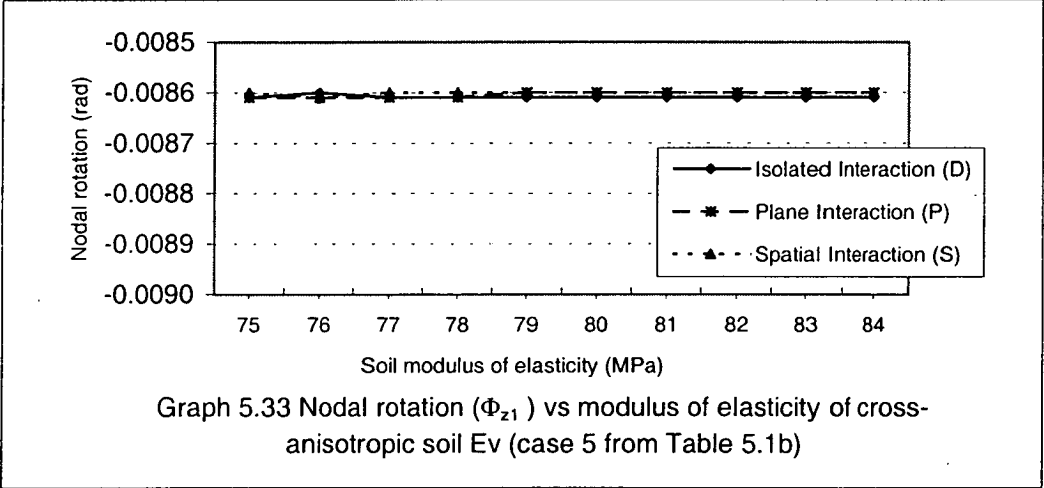
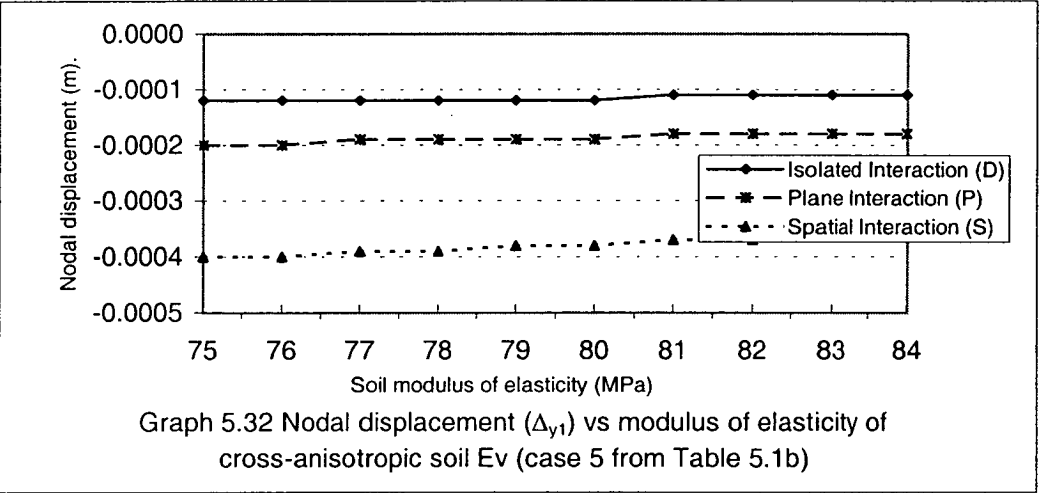
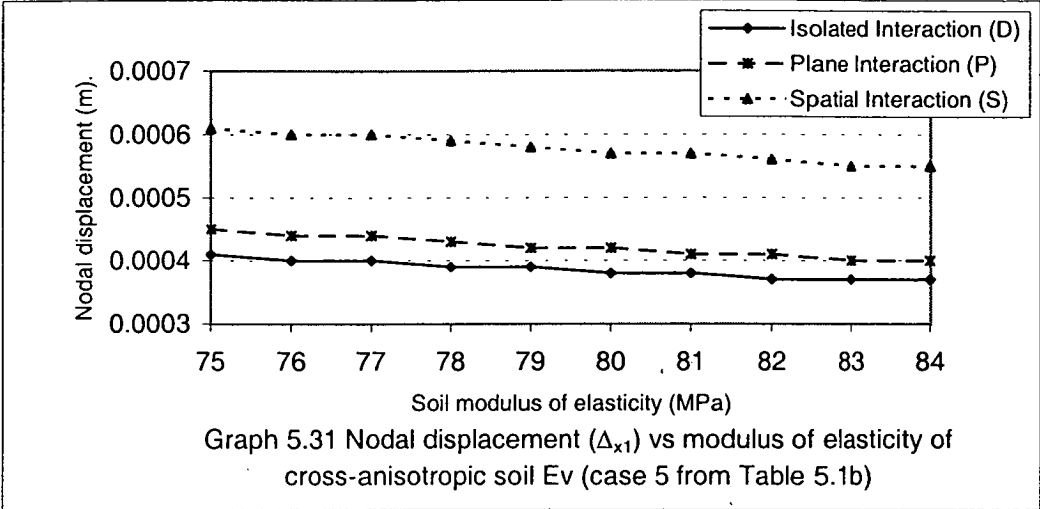




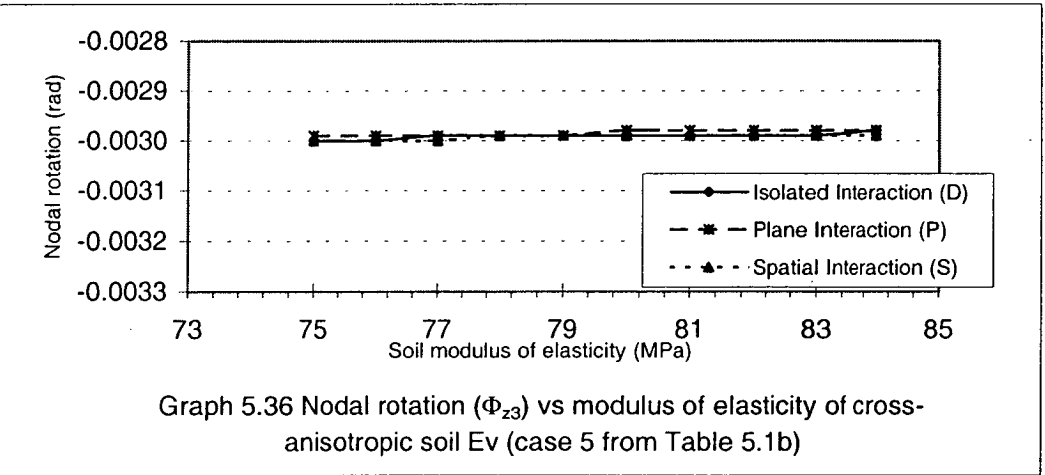
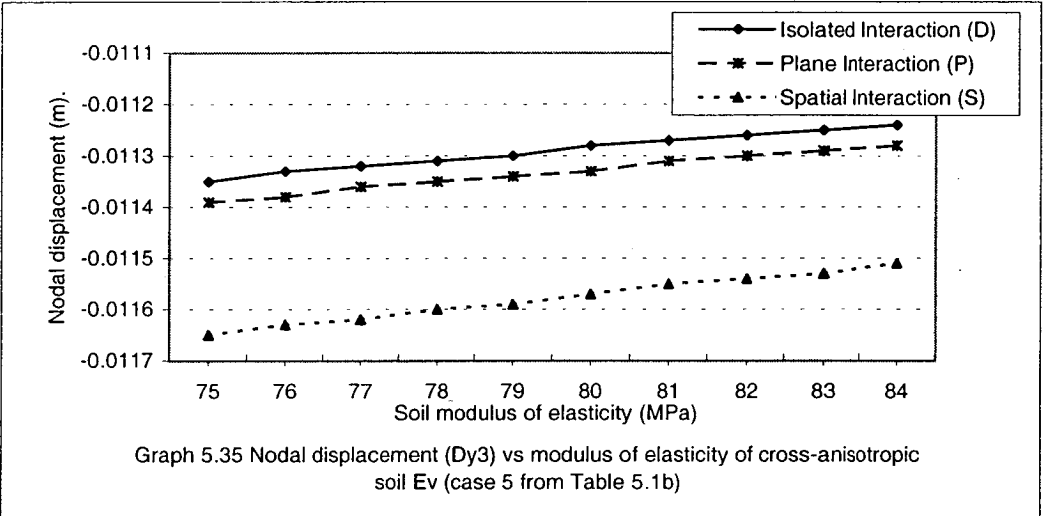
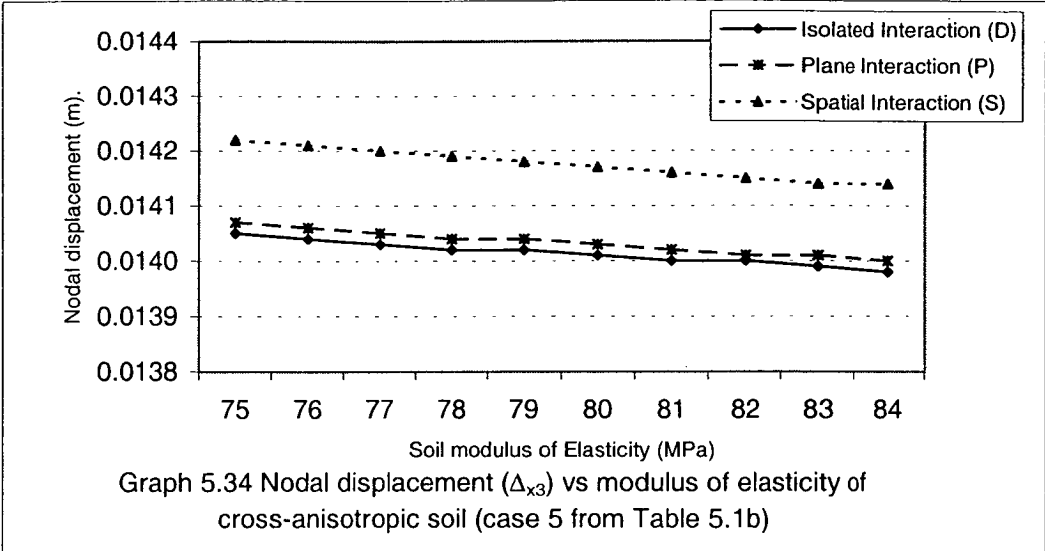
Graphs 5.25, 5.26 and 5.27 show variation of nodal displacements  $\Delta_{x3}$ ,  $\Delta_{y3}$  and nodal rotation  $\Phi_{z3}$  respectively at node 3 for  $E_v$  range between 26-35 MPa in cross-anisotropic iso soil (case 3 in Table 5.1b)



Graphs 5.28, 5.29 and 5.30 show variations in bending moments ( $M_{21}$ ,  $M_{32}$ ,  $M_{34}$ ) for members 1-2, 2-3 and 3-4 for variation in  $E_v$  from 75 to 84 MPa of cross-anisotropic soil (case 5 in Table 5.1b)



Graphs 5.31, 5.32 and 5.33 show variation of nodal displacements  $\Delta_{x1}$ ,  $\Delta_{y1}$  and nodal rotation  $\Phi_{z1}$  respectively at node 1 for  $E_v$  range between 75-84 MPa in cross-anisotropic iso soil (case 5 in Table 5.1b)



Graphs 5.34, 5.35 and 5.36 show variation of nodal displacements  $\Delta_{x3}$ ,  $\Delta_{y3}$  and nodal rotation  $\Phi_{z3}$  respectively at node 3 for  $E_v$  range between 75-84 MPa in cross-anisotropic iso soil (case 5 in Table 5.1b)

### 5.3.4. Set two Examples, Simulated Cases of Cross-anisotropic Soil

This section covers cases 7, 8, 9 and 10 (as shown in Tables 5.1 and 5.1c). These cases provide examples of cross-anisotropic soil. To investigate the structural internal forces and nodal displacements due to such a soil medium, one of the soil elastic parameters (i.e.  $F_v$ ) of four independent parameters is varied and the response of  $\beta^2$  and  $\alpha$  to such variations are reviewed.

In cases 7 and 8 which are typical of the natural medium, combination of the elastic parameters produces  $\beta^2$  positive and negative respectively. In case 9,  $\beta^2 = 0$  and  $\alpha \neq 1$  which is rather a rare combination of soil parameters to occur in nature. Whilst case 10 is a special configuration of cross-anisotropic soil where the elastic properties of soil represents isotropy in the medium.

The loaded structure is reproduced below, founded on a cross-anisotropic medium with properties mentioned in cases 7-10. These cases are presented in Table 5.1c.

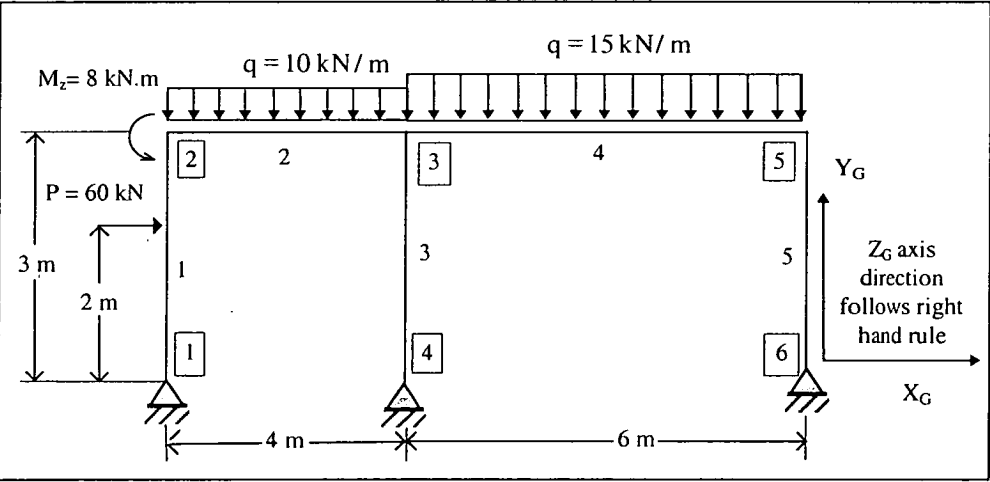
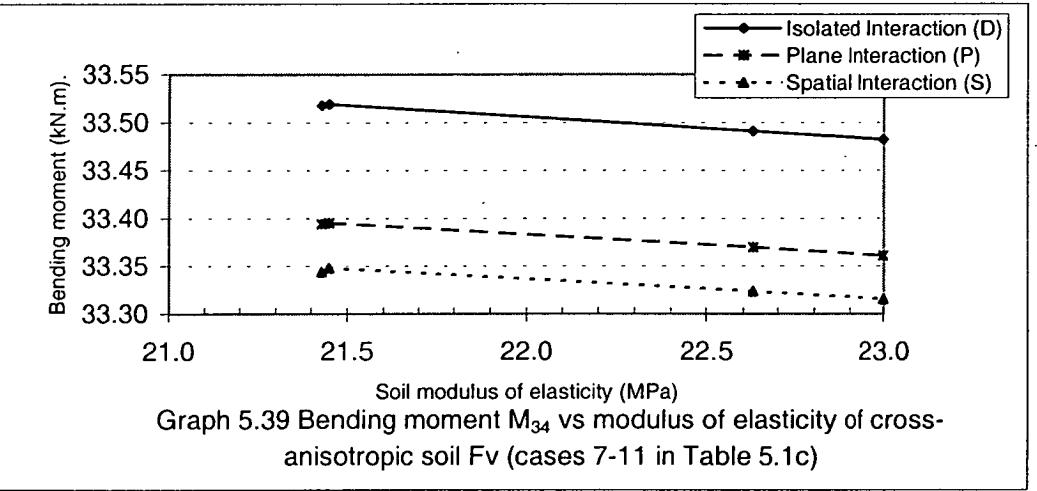
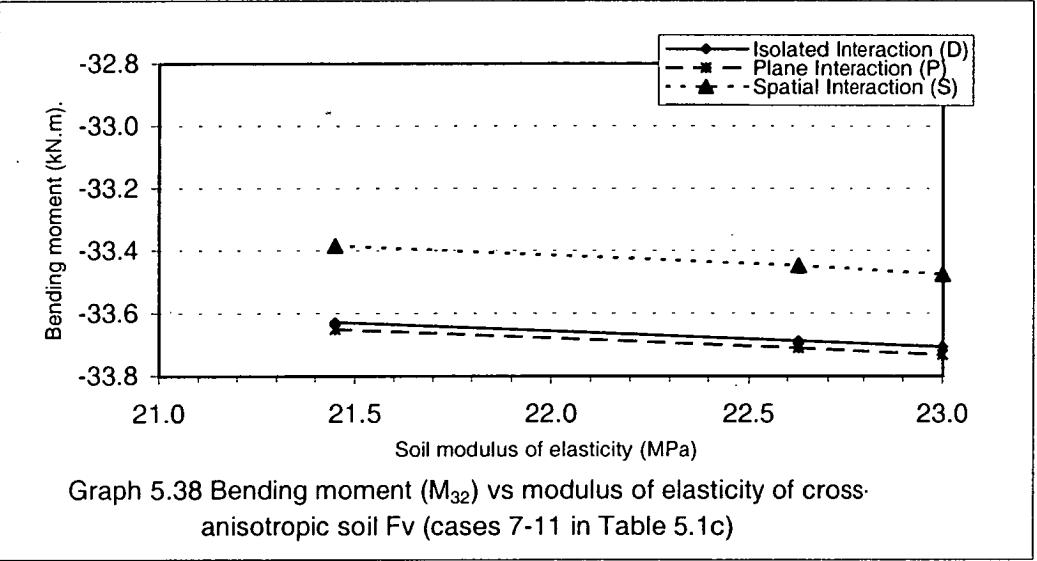
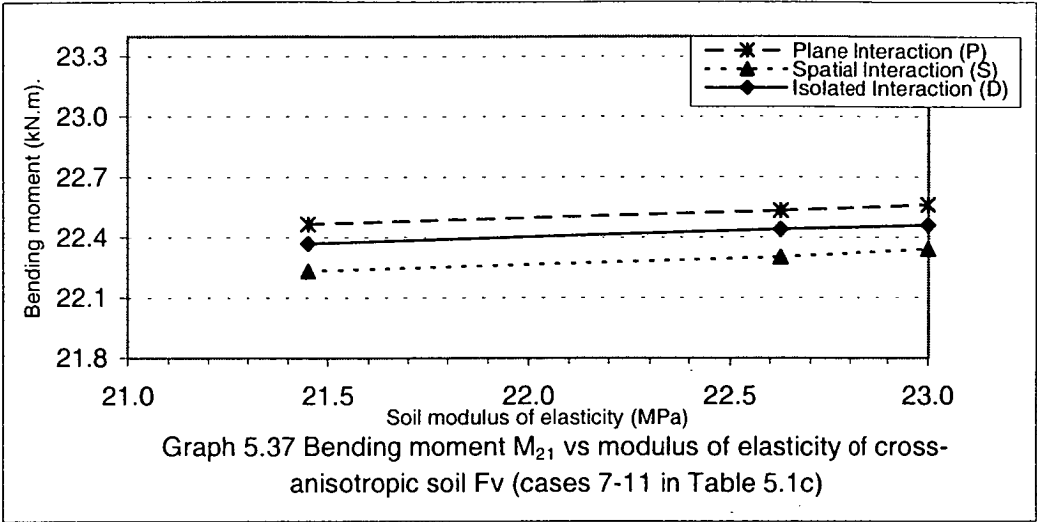
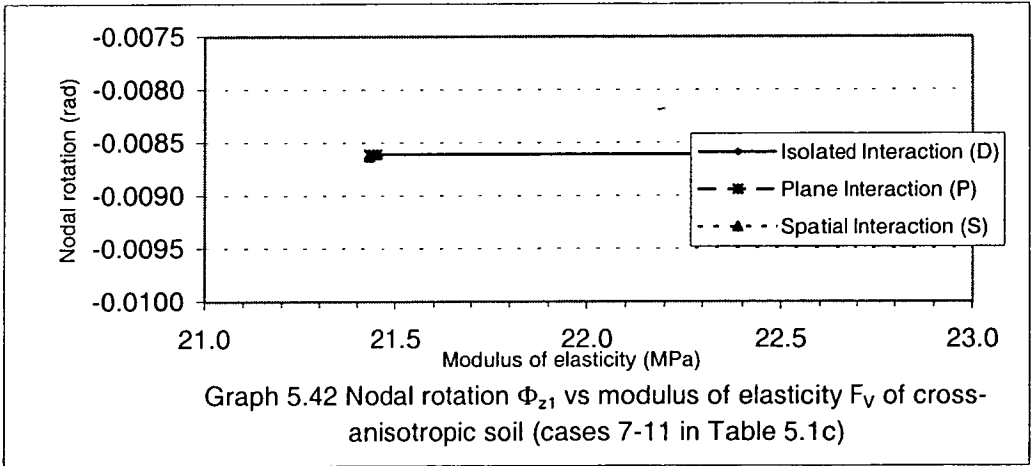
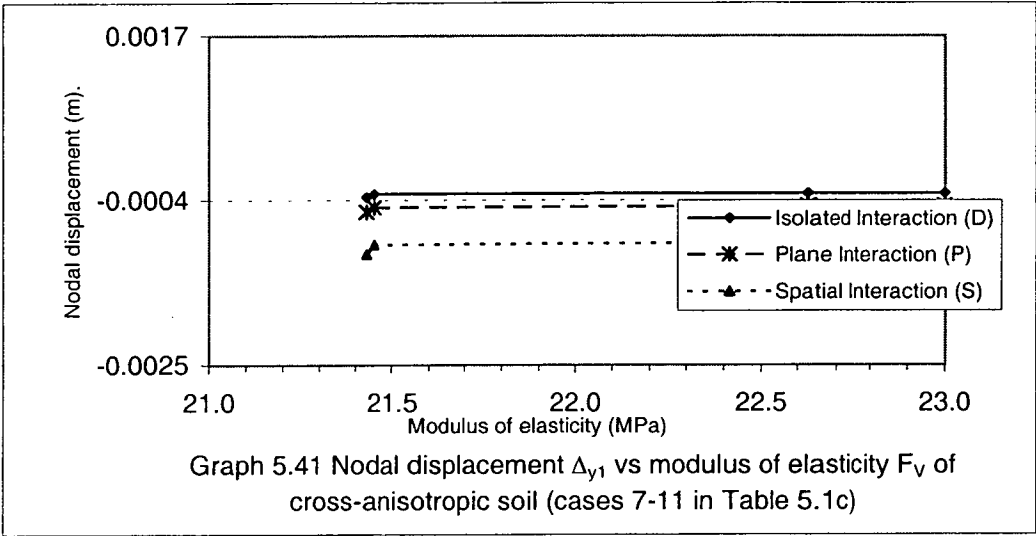
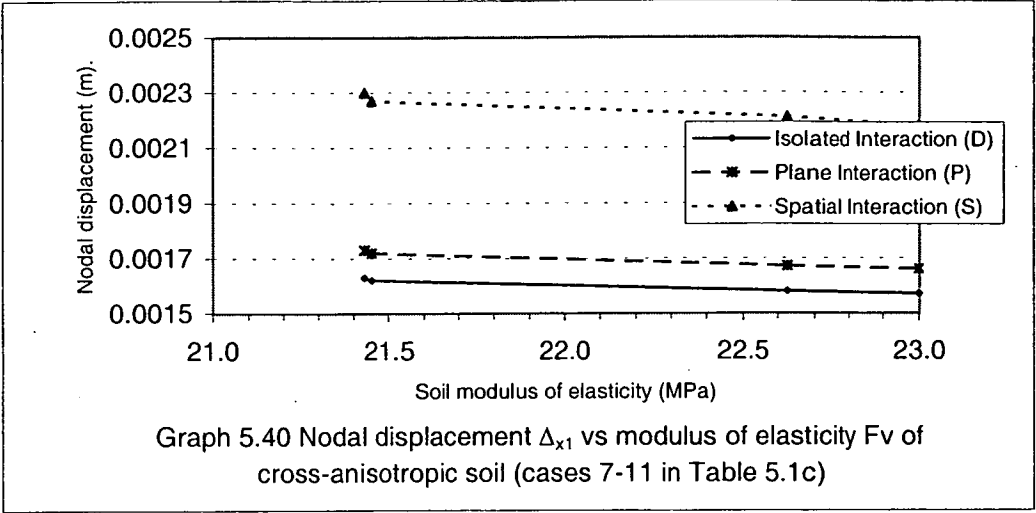


Figure 5.3 Profile of the structure and external applied loads

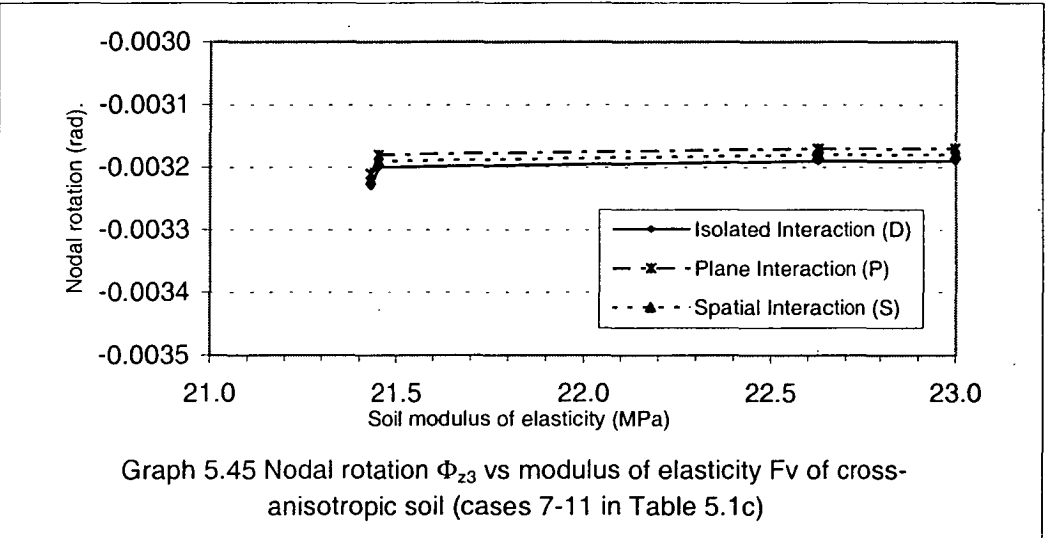
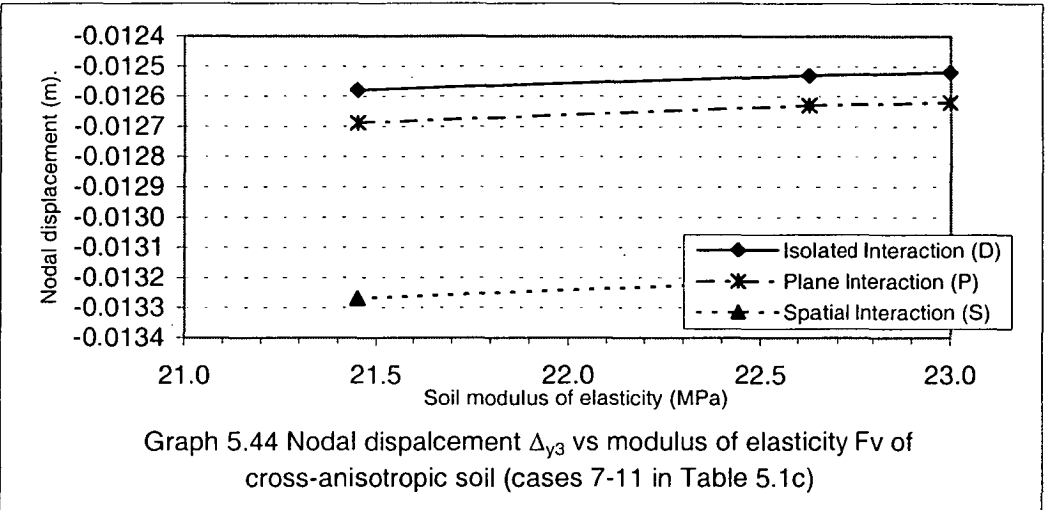
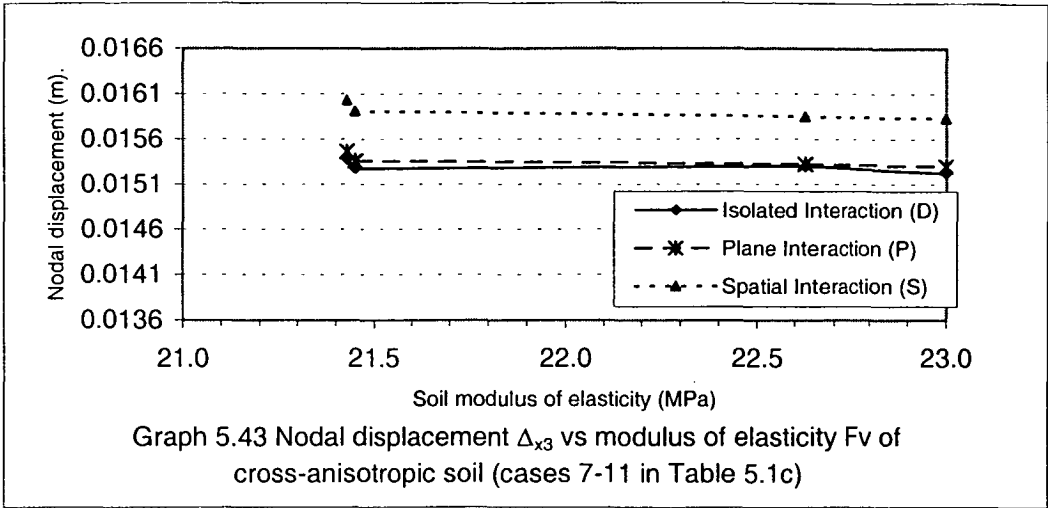
Graphs 5.37 to 5.45 present analysis output for element bending end-moments ( $M_{21}$ ,  $M_{33}$ ,  $M_{34}$ ) and nodal displacements ( $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$ ,  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$ ). Numerical values corresponding to these graphs are available in Appendix A5-3.



Graphs 5.37, 5.38 and 5.39 show variations in bending moments ( $M_{21}$ ,  $M_{32}$ ,  $M_{34}$ ) for members 1-2, 2-3 and 3-4 for variation in  $E_v$  from 21 to 23 MPa of cross-anisotropic soil (cases 7-11 in Table 5.1b)



Graphs 5.40, 5.41 and 5.42 show variation of nodal displacements  $\Delta_{x1}, \Delta_{y1}$  and nodal rotation  $\Phi_{z1}$  respectively at node 1 for  $E_v$  range between 21-23 MPa in cross-anisotropic soil (case 5 in Table 5.1b)



Graphs 5.43, 5.44 and 5.45 show variation of nodal displacements  $\Delta_{x3}$ ,  $\Delta_{y3}$  and nodal rotation  $\Phi_{z3}$  respectively at node 3 for  $F_v$  range between 21-23 MPa in cross-anisotropic soil (cases 7-11 in Table 5.1c)



### 5.4. STRUCTURE FOUNDED ON AN INFINITELY RIGID SOIL

For the purpose of comparison, the structure under the same load as that shown in Figure 5.3 is founded on infinitely rigid soil and the analysis is conducted using Space-Gass software (8.0, ITS 1999). The results for element bending moments, nodal displacements and support reactions are provided in Tables 5.3-5.5 respectively. Space-Gass is a frame analysis package widely used in structural design and analysis. This package has been extensively verified as producing data representative of true field performance (product users include BHP Engineering, CSIRO, Boral Johns Perry, Hydro Electric Commission, etc.).

Analysis results for the structure Z-dir hinge supported on an Infinitely rigid medium

Element	Location	Moment (kN.m)
1	M <sub>12</sub>	0.00E+00
	M <sub>21</sub>	2.56E+01
2	M <sub>23</sub>	1.76E+01
	M <sub>32</sub>	-3.66E+01
3	M <sub>34</sub>	-3.23E+01
	M <sub>43</sub>	0.00E+00
4	M <sub>35</sub>	-4.27E+00
	M <sub>53</sub>	-6.20E+01
5	M <sub>56</sub>	-6.20E+01
	M <sub>65</sub>	0.00E+00

Table 5.3 Element end moments

Node		Displacement (m) / Rotation(rad)
1	x	0.0000E+00
	y	0.0000E+00
	z	-8.6000E-03
2	x	1.9700E-02
	y	-1.0000E-03
	z	-3.6000E-03
3	x	1.3400E-02
	y	-1.0300E-02
	z	-2.9000E-03
4	x	0.0000E+00
	y	0.0000E+00
	z	-5.3000E-03
5	x	7.2000E-03
	y	-8.2000E-03
	z	7.0000E-04
6	x	0.0000E+00
	y	0.0000E+00
	z	-4.0000E-03

Table 5.4 Nodal displacements in x and y directions are in meters, and rotation in z direction is in radian

Node		Reaction
1	x	-2.85E+01
	y	6.44E+00
	z	0.00E+00
4	x	-1.08E+01
	y	6.89E+01
	z	0.00E+00
6	x	-2.07E+01
	y	5.46E+01
	z	0.00E+00

Table 5.5 Support reactions in x and y directions in kN, and for z direction is in kN.m

Tables 5.3, 5.4 & 5.5 present structure analysis outputs from Space Gass for hinge support frame on rigid medium.

### 5.5. STRUCTURE FOUNDED ON WINKLER SOIL

In this sub-section, Winkler spring model for the soil is considered. As presented earlier in Equation 5.3, when the settlement of a circular rigid footing on Winkler

medium is the same as that for an identical footing on the elastic half-space, the properties of the Winkler springs can be determined.

Considering the elastic constants in the isotropic elastic half-space case 12, 14 & 15 ( $E = 30 \text{ MPa}$  and  $\nu = 0.40, 0.15, 0.49$ ) in Table 5.1, where it is assumed that  $R = 0.50 \text{ m}$ , then the corresponding spring coefficient is obtained.

In the Tables 5.6 to 5.14 the following denotations are used:

- W: Winkler model  
 Iso: Isotropic soil idealisation  
 $k_s$ : Winkler spring coefficient  
 D, P, S: Isolated footing, plane frame (2D) and spatial frame interaction (3D)

The structure is then supported on these springs and is loaded with the same loads as shown in Figure 5.3. Space Gass software (8.0) is utilised and the results for element end bending moments, nodal displacements/ angular rotations and support reactions (shown in Tables 5.6 – 5.8) represent the analysis output for bending moments ( $M_{21}$ ,  $M_{32}$  &  $M_{34}$ ) versus isotropic modulus/ Winkler spring ( $k_s$ ) for soil data provided in Table 5.1d.

item	Win. Spring Coef. $k_s$	Bending moment in Winkler model	$\nu, E \text{ (kN/m}^2\text{)}$	Bending moment in Isotropic model (kN.m)		
	$\text{kN/m}^3$	$\text{kN.m}$		D	P	S
Wlso37	45496	3.06E+01	0.40, 30000	2.3052E+01	2.3147E+01	2.2859E+01
Wlso-1	39096	3.05E+01	0.15, 30000	2.2449E+01	2.2439E+01	2.1910E+01
Wlso-2	50292	3.07E+01	0.49, 30000	2.3386E+01	2.3530E+01	2.3364E+01

Table 5.6 Bending moment  $M_{21}$  for different Winkler springs and corresponding Isotropic values

item	Win. Spring Coef. $k_s$	Bending moment in Winkler model	$\nu, E \text{ (kN/m}^2\text{)}$	Bending moment in Isotropic model (kN.m)		
	$\text{kN/m}^3$	$\text{kN.m}$		D	P	S
Wlso37	45496	-3.32E+01	0.40, 30000	-3.26E+01	-3.26E+01	-3.23E+01
Wlso-1	39096	-3.30E+01	0.15, 30000	-3.22E+01	-3.22E+01	-3.18E+01
Wlso-2	50292	-3.33E+01	0.49, 30000	-3.29E+01	-3.29E+01	-3.26E+01

Table 5.7 Bending moment  $M_{32}$  for different Winkler springs and corresponding Isotropic values

item	Win. Spring Coef. $k_s$	Bending moment in Winkler model	$\nu, E \text{ (kN/m}^2\text{)}$	Bending moment in Isotropic model (kN.m)		
	$\text{kN/m}^3$	$\text{kN.m}$		D	P	S
Wlso37	45496	-2.85E+01	0.40, 30000	3.2391E+01	3.2259E+01	3.2210E+01
Wlso-1	39096	-2.85E+01	0.15, 30000	3.2685E+01	3.2580E+01	3.2519E+01
Wlso-2	50292	-2.85E+01	0.49, 30000	3.2251E+01	3.2108E+01	3.2066E+01

Table 5.8 Bending moment  $M_{34}$  for different Winkler springs and corresponding Isotropic values

Tables 5.9 – 5.11 represent the analysis output for nodal displacements ( $\Delta_{x1}$ ,  $\Delta_{y1}$  &  $\Phi_{z1}$ ) versus isotropic modulus/ Winkler spring ( $k_s$ ) for the soil data provided in Table 5.1d.

item	Win. Spring Coef. $k_s$	Nodal displacement $\Delta_{x1}$ in Winkler model	$\nu, E \text{ (kN/m}^2\text{)}$	Nodal displacement $\Delta_{x1}$ in Isotropic model (m)		
	$\text{kN/m}^3$	m		D	P	S
WIso37	45496	0.0000E+00	0.40, 30000	2.0000E-04	3.0000E-04	9.0000E-04
WIso-1	39096	0.0000E+00	0.15, 30000	6.0000E-04	7.0000E-04	1.4000E-03
WIso-2	50292	0.0000E+00	0.49, 30000	0.0000E+00	1.0000E-04	6.0000E-04

Table 5.9 Nodal displacement  $\Delta_{x1}$  for different Winkler springs and the corresponding Isotropic values

item	Win. Spring Coef. $k_s$	Nodal displacement $\Delta_{y1}$ in Winkler model	$\nu, E \text{ (kN/m}^2\text{)}$	Nodal displacement $\Delta_{y1}$ in Isotropic model (m)		
	$\text{kN/m}^3$	m		D	P	S
WIso37	45496	-3.00E-04	0.40, 30000	-4.0000E-04	-6.0000E-04	-1.1000E-03
WIso-1	39096	-3.00E-04	0.15, 30000	-4.0000E-04	-6.0000E-04	-1.2000E-03
WIso-2	50292	-2.00E-04	0.49, 30000	-3.0000E-04	-5.0000E-04	-1.0000E-03

Table 5.10 Nodal displacement  $\Delta_{y1}$  for different Winkler springs and the corresponding isotropic values

item	Win. Spring Coef. $k_s$	Nodal rotation $\Phi_{z1}$ in Winkler model	$\nu, E \text{ (kN/m}^2\text{)}$	Nodal rotation $\Phi_{z1}$ in Isotropic model (rad)		
	$\text{kN/m}^3$	rad		D	P	S
WIso37	45496	-1.68E-02	0.40, 30000	-8.8000E-03	-8.8000E-03	-8.8000E-03
WIso-1	39096	-1.69E-02	0.15, 30000	-8.8000E-03	-8.8000E-03	-8.8000E-03
WIso-2	50292	-1.68E-02	0.49, 30000	-8.8000E-03	-8.8000E-03	-8.8000E-03

Table 5.11 Nodal rotation  $\Phi_{z1}$  for different Winkler springs and the corresponding Isotropic values

Tables 5.12 – 5.14 represent the analysis output for nodal displacements ( $\Delta_{x3}$ ,  $\Delta_{y3}$  &  $\Phi_{z3}$ ) versus isotropic modulus/ Winkler spring ( $k_s$ ) for the soil data provided in Table 5.1d.

item	Win. Spring Coef. $k_s$	Nodal displacement $\Delta_{x3}$ in Winkler model	$\nu, E \text{ (kN/m}^2\text{)}$	Nodal displacement $\Delta_{x3}$ in Isotropic model (m)		
	$\text{kN/m}^3$	m		D	P	S
WIso37	45496	2.6000E-02	0.40, 30000	2.00E-04	3.00E-04	9.00E-04
WIso-1	39096	2.6100E-02	0.15, 30000	6.00E-04	7.00E-04	1.40E-03
WIso-2	50292	2.6000E-02	0.49, 30000	0.00E+00	1.00E-04	6.00E-04

Table 5.12 Nodal Displacement  $\Delta_{x3}$  for different Winkler springs and the corresponding isotropic

item	Win. Spring Coef. $k_s$	Nodal displacement $\Delta_{y3}$ in Winkler model	$\nu, E \text{ (kN/m}^2\text{)}$	Nodal displacement $\Delta_{y3}$ in Isotropic model (m)		
	$\text{kN/m}^3$	m		D	P	S
WIso37	45496	-2.1700E-02	0.40, 30000	-1.2900E-02	-1.3000E-02	-1.3700E-02
WIso-1	39096	-2.1900E-02	0.15, 30000	-1.3300E-02	-1.3400E-02	-1.4200E-02
WIso-2	50292	-2.1500E-02	0.49, 30000	-1.2600E-02	-1.2800E-02	-1.3400E-02

Table 5.13 Nodal Displacement  $\Delta_{y3}$  for different Winkler springs and the corresponding isotropic values

item	Win. Spring Coef. $k_s$	Nodal rotation $\Phi_{23}$ in Winkler model	$\nu, E$ (kN/m <sup>2</sup> )	Nodal rotation $\Phi_{23}$ in Isotropic model (rad)		
	kN/m <sup>3</sup>	rad		D	P	S
WIso37	45496	-5.80E-03	0.40, 30000	-3.2000E-03	-3.1000E-03	-3.2000E-03
WIso-1	39096	-5.80E-03	0.15, 30000	-3.2000E-03	-3.2000E-03	-3.2000E-03
WIso-2	50292	-5.80E-03	0.49, 30000	-3.1000E-03	-3.1000E-03	-3.1000E-03

Table 5.14 Nodal rotation  $\Phi_{23}$  for different Winkler springs and the corresponding isotropic values

## 5.6. VERIFICATION OF RESULTS OF ANALYSIS

Since the Direct Stiffness Matrix Method is a well-known approach for analysing structures and mainly used in computer programs, verification of this method is aimless. However, effort is made towards the verification of results of analyses.

To verify the results of the interaction analysis, the frame, which is subjected to the same external load stipulated in Figure 5.3 and the support displacements (as prescribed displacement) obtained from *SASIAP*, is considered. This frame is then analysed by Space Gass software (8.0, ITS 1999) which is a widely used industry standard package (I.T.S. Integrated Technical Softwares Pty. Ltd. 1999).

The reactions from the interaction analysis were applied to the half-space and the appropriate Boussinesq and Cerruti relations for isotropic soil were employed (by using the corresponding soil flexibility matrix), while the Gerrard, Wardle and Harrison force-displacement relations in soil were used for cross-anisotropic. For this verification, Microsoft Excel (1997) was used to check the support displacements obtained from the interaction analysis and the results were compared to those from Space Gass when the frame is subjected to the support displacements obtained from the force-displacement relationships. Good agreement of the corresponding results testified to the adequacy of the developed stiffness matrix of the elastic half-space. An example associated with result verification is shown in Tables 5.15 to 5.23, and a diagrammatic representation is provided Graphs 5.46 to 5.48.

Tables 5.21 to 5.23 present numerical values for verification for an isotropic sub-case ( $E_s = 50 \text{ MPa}$ ,  $\nu = 0.25$ ) for three interaction modes  $D$ ,  $P$  and  $S$ . Graphs 5.46 to

5.48 are reproduction of Graphs 5.1 to 5.3 that are used to show the points for which the verification of results is conducted.

## **5.7. JUSTIFICATION OF THE TYPE OF FRAME AND LOADS USED IN THE ANALYSES**

Frames are a popular type of superstructures that are regularly considered in structural design and construction. In this group of structures, depending on the demand and architecture of design, planar frames with two to three unequal bays mostly are found in industrial and residential buildings. These frames are subjected to different type of load and more frequently considered loads are gravity and wind loads. Due to design, construction, structure response, and frequency of applications, connection of frame columns to their footing (isolated footings) are considered hinged. This option of course is an alternative to a fixed connection that is used in continuous footings (beam foundations).

Briefly, as a typical structure, an unequal width two-bay planar single storey hinged support with a composition of loads such as different vertically distributed loads, a horizontal concentrated force and a concentrated moment (Figure 5.3) is taken as an example of an inclusive example for the analysis used in the project.

A particular load case and structure geometry (see Figure 5.3) is considered to illustrate the general analysis technique used to investigate the soil-structure interaction.

# Reactions resulting from interaction analysis

## Structure hinged to support in z-direction

Reaction	Infinitely Rigid	7-IsoD	7-IsoP	7-IsoS
$R_{x1}$	-28.5385	-27.9160	-27.9226	-27.8278
$R_{y1}$	6.4433	7.5906	7.5809	7.7130
$R_{x4}$	-10.7806	-10.8284	-10.8051	-10.7933
$R_{y4}$	68.9279	67.0157	67.0318	66.8117
$R_{x6}$	-20.6809	-21.2557	-21.2723	-21.3789
$R_{y6}$	54.6288	55.3937	55.3873	55.4753

Table 5.15 Nodal reactions from interaction analysis (Isotropic  $E_s=50$  MPa,  $\nu=0.25$ )

## End-member bending moments resulting from interaction analysis ( structure is hinged to support in z-direction)

	Infinitely Rigid	7-IsoD	7-IsoP	7-IsoS
$M_{21}$	25.6155	23.4834	23.7478	23.7679
$M_{23}$	-17.6155	-15.4834	-15.7478	-15.7679
$M_{32}$	-36.6114	-33.6645	-33.8898	-33.9083
$M_{34}$	32.3418	32.3798	32.4851	32.4152
$M_{35}$	4.2697	1.2847	1.4047	1.4931
$M_{53}$	-62.0427	-64.1368	-63.7671	-63.8169
$M_{56}$	62.0427	64.1368	63.7671	63.8169

Table 5.16 Members end moments (kN.m) from interaction analysis (Isotropic soil  $E_s=50$  MPa,  $\nu=0.25$ )

## Nodal displacements / Nodal rotations resulting from interaction analysis

### Structure is hinged to support in z-direction

	Infinitely Rigid	7-IsoD	7-IsoP	7-IsoS
$\Delta_{x1}$	0.00000	0.00028	0.00036	0.00074
$\Delta_{y1}$	0.00000	-0.00023	-0.00035	-0.00069
$\Phi_{z1}$	-0.00855	-0.00871	-0.00870	-0.00868
$\Delta_{x2}$	0.01973	0.02062	0.02067	0.02103
$\Delta_{y2}$	-0.00097	-0.00137	-0.00149	-0.00185
$\Phi_{z2}$	-0.00363	-0.00393	-0.00391	-0.00392
$\Delta_{x3}$	0.01344	0.01420	0.01425	0.01459
$\Delta_{y3}$	-0.01034	-0.01205	-0.01213	-0.01257
$\Phi_{z3}$	-0.00286	-0.00307	-0.00306	-0.00307
$\Delta_{x4}$	0.00000	0.00011	0.00021	0.00051
$\Delta_{y4}$	0.00000	-0.00200	-0.00207	-0.00255
$\Phi_{z4}$	-0.00529	-0.00551	-0.00549	-0.00550
$\Delta_{x5}$	0.00724	0.00782	0.00787	0.00818
$\Delta_{y5}$	-0.00819	-0.00996	-0.01004	-0.01053
$\Phi_{z5}$	0.00069	0.00065	0.00064	0.00061
$\Delta_{x6}$	0.00000	0.00021	0.00022	0.00040
$\Delta_{y6}$	0.00000	-0.00165	-0.00173	-0.00220
$\Phi_{z6}$	-0.00396	-0.00413	-0.00414	-0.00420

Table 5.17 Nodal displacements (m) Angle of rotations (rad) from interaction analysis (Isotropic soil  $E_s=50$  MPa,  $\nu=0.25$ )

Verification for Input: 7-IsoD.txt (Isolated footing Interaction D)

REACTIONS FROM THE INTERACTION ANALYSIS (SASIAP, 7-IsoD.txt) are applied to the corresponding Soil Flexi Matrix (diagonal mode). Then, the related transverse displacements are individually calculated.								
NODE # 1			NODE # 4			NODE # 6		
H-reaction	V-reaction	M-reaction	H-reaction	V-reaction	M-reaction	H-reaction	V-reaction	M-reaction
-2.79160E+01	7.59060E+00	0.00000E+00	-1.08284E+01	6.70157E+01	0.00000E+00	-2.12557E+01	5.53937E+01	0.00000E+00
Flexibility Coef's for Soil members (DISPLACEMENT AT DOF(j) DUE TO UNIT FORCE APPLIED TO THE DOF(i)) :								
9.95000E-06	0.00000E+00	-2.48700E-05	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	2.98400E-05	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
-2.48700E-05	0.00000E+00	7.46040E-04	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	9.95000E-06	0.00000E+00	-2.48700E-05	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	2.98400E-05	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	-2.48700E-05	0.00000E+00	7.46040E-04	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	9.95000E-06	0.00000E+00	-2.48700E-05
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	2.98400E-05	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	-2.48700E-05	0.00000E+00	7.46040E-04
TOTAL DISPLACEMENT								
-2.77764E-04	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	2.26504E-04	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
6.94270E-04	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	-1.07742E-04	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	1.99975E-03	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	2.69302E-04	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	-2.11494E-04	0.00000E+00	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	1.65295E-03	0.00000E+00
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	5.28629E-04	0.00000E+00	0.00000E+00

A

B

Table 5.18 Verification of displacement for sample: 7-1Iso D

**Note:** In section A, the support reactions from the interaction analysis (SASIAP) are associated with the corresponding soil flexibility matrix (for Isolated case). In section B, quantities in vertical direction are obtained from multiplication of corresponding cells from section A. Finally, the end column in section B is produced from the summation of the individual sub-displacements of section B.

Verification for Input: 7-IsoP.txt (Plane frame Interaction P)

REACTIONS FROM THE INTERACTION ANALYSIS (SASIAP, 7-IsoP.txt) are applied to the corresponding Soil Flexi Matrix (Plane mode). Then, the related transverse Displacements are individually calculated.									
NODE # 1			NODE # 4			NODE # 6			
H-reaction	V-reaction	M-reaction	H-reaction	V-reaction	M-reaction	H-reaction	V-reaction	M-reaction	
-2.79226E+01	7.58094E+00	0.00000E+00	-1.08051E+01	6.70318E+01	0.00000E+00	-2.12723E+01	5.53873E+01	0.00000E+00	A
Flexibility Coef's for Soil members (DISPLACEMENT AT DOF(j)) DUE TO UNIT FORCE APPLIED TO THE DOF(i)) :									
9.9500E-06	0.0000E+00	-2.4870E-05	1.9900E-06	-5.0000E-07	1.2000E-07	8.0000E-07	-2.0000E-07	2.0000E-08	
0.0000E+00	2.9840E-05	0.0000E+00	5.0000E-07	1.4900E-06	-3.7000E-07	2.0000E-07	6.0000E-07	-6.0000E-08	
-2.4870E-05	0.0000E+00	7.4604E-04	1.2000E-07	3.7000E-07	-1.9000E-07	2.0000E-08	6.0000E-08	-1.0000E-08	
1.9900E-06	5.0000E-07	1.2000E-07	9.9500E-06	0.0000E+00	-2.4870E-05	1.3300E-06	-3.3000E-07	6.0000E-08	
-5.0000E-07	1.4900E-06	3.7000E-07	0.0000E+00	2.9840E-05	0.0000E+00	3.3000E-07	9.9000E-07	-1.7000E-07	
1.2000E-07	-3.7000E-07	-1.9000E-07	-2.4870E-05	0.0000E+00	7.4604E-04	6.0000E-08	1.7000E-07	-6.0000E-08	
8.0000E-07	2.0000E-07	2.0000E-08	1.3300E-06	3.3000E-07	6.0000E-08	9.9500E-06	0.0000E+00	-2.4870E-05	
-2.0000E-07	6.0000E-07	6.0000E-08	-3.3000E-07	9.9000E-07	1.7000E-07	0.0000E+00	2.9840E-05	0.0000E+00	
2.0000E-08	-6.0000E-08	-1.0000E-08	6.0000E-08	-1.7000E-07	-6.0000E-08	-2.4870E-05	0.0000E+00	7.4604E-04	
								TOTAL DISPLACEMENT	
-2.77830E-04	0.00000E+00	0.00000E+00	-2.15021E-05	-3.35159E-05	0.00000E+00	-1.70178E-05	-1.10775E-05	0.00000E+00	-3.60944E-04
0.00000E+00	2.26215E-04	0.00000E+00	-5.40254E-06	9.98773E-05	0.00000E+00	-4.25446E-06	3.32324E-05	0.00000E+00	3.49668E-04
6.94436E-04	0.00000E+00	0.00000E+00	-1.29661E-06	2.48018E-05	0.00000E+00	-4.25446E-07	3.32324E-06	0.00000E+00	7.20839E-04
-5.55661E-05	3.79047E-06	0.00000E+00	-1.07510E-04	0.00000E+00	0.00000E+00	-2.82921E-05	-1.82778E-05	0.00000E+00	-2.05856E-04
1.39613E-05	1.12956E-05	0.00000E+00	0.00000E+00	2.00023E-03	0.00000E+00	-7.01986E-06	5.48334E-05	0.00000E+00	2.07330E-03
-3.35072E-06	-2.80495E-06	0.00000E+00	2.68722E-04	0.00000E+00	0.00000E+00	-1.27634E-06	9.41584E-06	0.00000E+00	2.70706E-04
-2.23381E-05	1.51619E-06	0.00000E+00	-1.43707E-05	2.21205E-05	0.00000E+00	-2.11659E-04	0.00000E+00	0.00000E+00	-2.24731E-04
5.58453E-06	4.54856E-06	0.00000E+00	3.56567E-06	6.63615E-05	0.00000E+00	0.00000E+00	1.65276E-03	0.00000E+00	1.73282E-03
-5.58453E-07	-4.54856E-07	0.00000E+00	-6.48304E-07	-1.13954E-05	0.00000E+00	5.29042E-04	0.00000E+00	0.00000E+00	5.15985E-04
B									

Table 5.19 Verification of displacement for sample: 7-1Iso P

**Note:** In section **A**, the support reactions from the interaction analysis (SASIAP) are associated with the corresponding soil flexibility matrix (for Plane frame case). In section **B**, quantities in vertical direction are obtained from multiplication of corresponding cells from section **A**. Finally, the end column in section **B** is produced from the summation of the individual sub-displacements of section **B**.

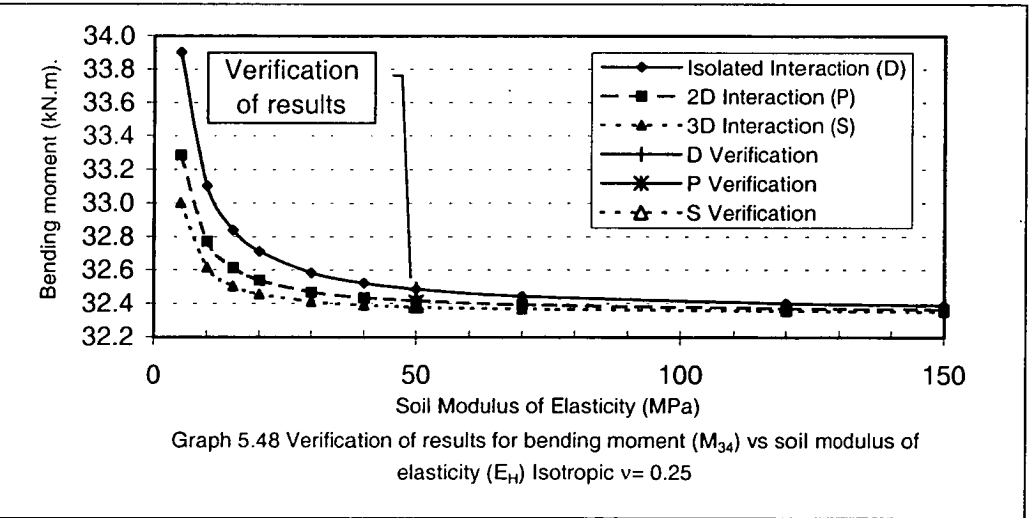
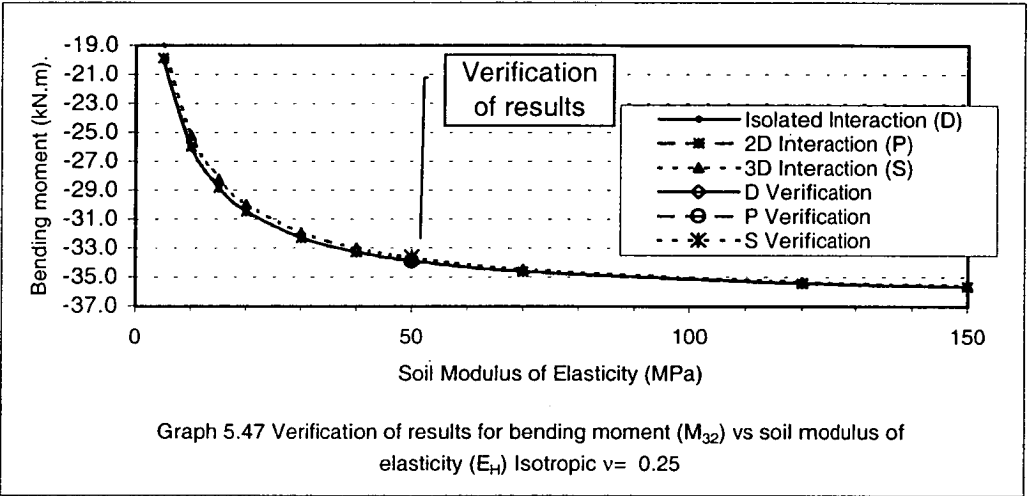
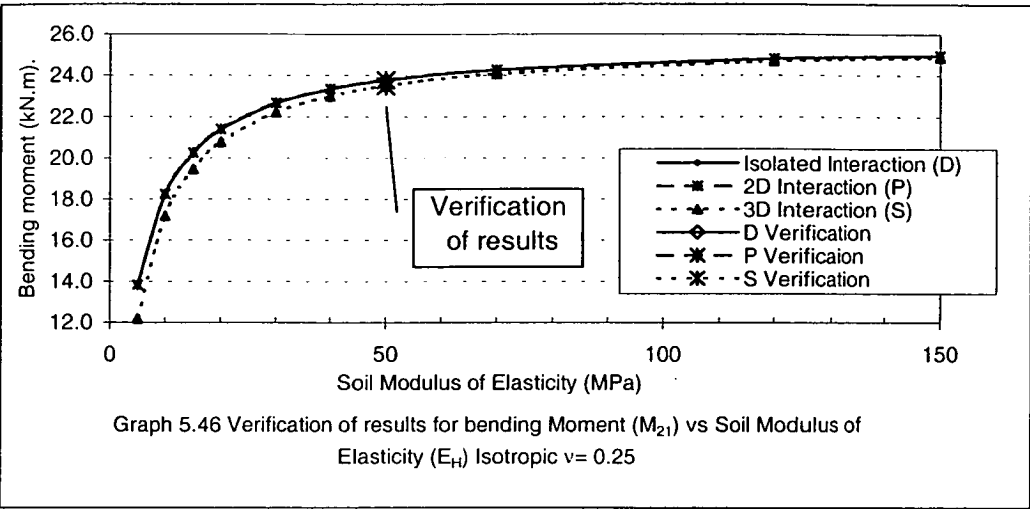


### Verification for Input: 7-IsoS.txt (Space frame Interaction S)

REACTIONS FROM THE INTERACTION ANALYSIS (SASIAP, 7-IsoS.txt) are applied to the corresponding Soil Flexi Matrix (Space mode). Then, the related transverse Displacements are individually calculated.								
NODE # 1			NODE # 4			NODE # 6		
H-reaction	V-reaction	M-reaction	H-reaction	V-reaction	M-reaction	H-reaction	V-reaction	M-reaction
-2.78278E+01	7.71301E+00	0.00000E+00	-1.07933E+01	6.68117E+01	0.00000E+00	-2.13789E+01	5.54753E+01	0.00000E+00
Flexibility Coef's for Soil members (DISPLACEMENT AT DOF(j) DUE TO UNIT FORCE APPLIED TO THE DOF(i)) :								
1.5720E-05	0.0000E+00	-2.4570E-05	6.4300E-06	-1.6100E-06	2.9000E-07	3.4600E-06	-8.6000E-07	8.0000E-08
0.0000E+00	3.4170E-05	0.0000E+00	1.6100E-06	4.8200E-06	-8.7000E-07	8.6000E-07	2.5900E-06	-2.3000E-07
-2.4570E-05	0.0000E+00	7.4563E-04	2.9000E-07	8.7000E-07	-3.4000E-07	8.0000E-08	2.3000E-07	-4.0000E-08
6.4300E-06	1.6100E-06	2.9000E-07	1.5720E-05	0.0000E+00	-2.4570E-05	5.0100E-06	-1.2500E-06	1.7000E-07
-1.6100E-06	4.8200E-06	8.7000E-07	0.0000E+00	3.4170E-05	0.0000E+00	1.2500E-06	3.7600E-06	-5.0000E-07
2.9000E-07	-8.7000E-07	-3.4000E-07	-2.4570E-05	0.0000E+00	7.4563E-04	1.7000E-07	5.0000E-07	-1.4000E-07
3.4600E-06	8.6000E-07	8.0000E-08	5.0100E-06	1.2500E-06	1.7000E-07	1.5720E-05	0.0000E+00	-2.4570E-05
-8.6000E-07	2.5900E-06	2.3000E-07	-1.2500E-06	3.7600E-06	5.0000E-07	0.0000E+00	3.4170E-05	0.0000E+00
8.0000E-08	-2.3000E-07	-4.0000E-08	1.7000E-07	-5.0000E-07	-1.4000E-07	-2.4570E-05	0.0000E+00	7.4563E-04
								TOTAL DISPLACEMENT
-4.3745E-04	0.0000E+00	0.0000E+00	-6.9401E-05	-1.0757E-04	0.0000E+00	-7.3971E-05	-4.7709E-05	0.0000E+00
0.0000E+00	2.6355E-04	0.0000E+00	-1.7377E-05	3.2203E-04	0.0000E+00	-1.8386E-05	1.4368E-04	0.0000E+00
6.8373E-04	0.0000E+00	0.0000E+00	-3.1300E-06	5.8126E-05	0.0000E+00	-1.7103E-06	1.2759E-05	0.0000E+00
-1.7893E-04	1.2418E-05	0.0000E+00	-1.6967E-04	0.0000E+00	0.0000E+00	-1.0711E-04	-6.9344E-05	0.0000E+00
4.4803E-05	3.7177E-05	0.0000E+00	0.0000E+00	2.2830E-03	0.0000E+00	-2.6724E-05	2.0859E-04	0.0000E+00
-8.0701E-06	-6.7103E-06	0.0000E+00	2.6519E-04	0.0000E+00	0.0000E+00	-3.6344E-06	2.7738E-05	0.0000E+00
-9.6284E-05	6.6332E-06	0.0000E+00	-5.4074E-05	8.3515E-05	0.0000E+00	-3.3608E-04	0.0000E+00	0.0000E+00
2.3932E-05	1.9977E-05	0.0000E+00	1.3492E-05	2.5121E-04	0.0000E+00	0.0000E+00	1.8956E-03	0.0000E+00
-2.2262E-06	-1.7740E-06	0.0000E+00	-1.8349E-06	-3.3406E-05	0.0000E+00	5.2528E-04	0.0000E+00	0.0000E+00

Table 5.20 Verification of displacement for sample: 7-1Iso S

**Note:** In section **A**, the support reactions from the interaction analysis (SASIAP) are associated with the corresponding soil flexibility matrix (for Space frame case). In section **B**, quantities in vertical direction are obtained from multiplication of corresponding cells from section **A**. Finally, the end column in section **B** is produced from the summation of the individual sub-displacements of section **B**.



Graphs 5.46, 5.47 and 5.48 show verification of results for the bending moment in the members 2-1, 2-3 and 3-4 ( $M_{21}$ ,  $M_{32}$  and  $M_{34}$ ) respectively for isotropic soil with  $\nu=0.25$  (See Table 5.1a)

### Bending Moment ( $M_{32}$ and $M_{34}$ ) vs Isotropic soil Modulus of Elasticity ( $E_H$ )

(See Table 5.1a)  $\nu=0.25$

	MPa	kN.m	kN.m	kN.m	kN.m	kN.m	kN.m
Sub-case	E soil	Verif. D	D group	Verif. P	P group	Verif. S	S group
1-Iso	5		13.8708		13.8171		12.1923
2-Iso	10		18.2303		18.2473		17.1872
3-Iso	15		20.2241		20.2531		19.4632
4-Iso	20		21.3693		21.3996		20.7698
5-Iso	30		22.6349		22.6619		22.2136
6-Iso	40		23.3191		23.3423		22.9942
7-Iso	50	<b>23.7468</b>	23.7478	<b>23.7735</b>	23.7679	<b>23.4699</b>	23.4834
8-Iso	70		24.2556		24.2712		24.0629
9-Iso	120		24.8058		24.8158		24.6910
10-Iso	150		24.9640		24.9722		24.8716

Table 5.21 Bending moment  $M_{21}$  vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

	MPa	kN.m	kN.m	kN.m	kN.m	kN.m	kN.m
Sub-case	E soil	Verif. D	D group	Verif. P	P group	Verif. S	S group
1-Iso	5		-20.0225		-19.8812		-18.9020
2-Iso	10		-26.0066		-25.9861		-25.2084
3-Iso	15		-28.8234		-28.8328		-28.2145
4-Iso	20		-30.4588		-30.4774		-29.9682
5-Iso	30		-32.2790		-32.3009		-31.9264
6-Iso	40		-33.2683		-33.2888		-32.9933
7-Iso	50	<b>-33.8892</b>	-33.8898	<b>-33.9073</b>	-33.9083	<b>-33.6535</b>	-33.6645
8-Iso	70		-34.6275		-34.6426		-34.4621
9-Iso	120		-35.4288		-35.4389		-35.3295
10-Iso	150		-35.65952		-35.66788		-35.57937

Table 5.22 Bending moment  $M_{32}$  vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

	MPa	kN.m	kN.m	kN.m	kN.m	kN.m	kN.m
Sub-case	E soil	Verif. D	D group	Verif. P	P group	Verif. S	S group
1-Iso	5		33.9010		33.2873		33.0020
2-Iso	10		33.1011		32.7724		32.6155
3-Iso	15		32.8389		32.6142		32.5049
4-Iso	20		32.7102		32.5396		32.4555
5-Iso	30		32.5839		32.4687		32.4110
6-Iso	40		32.5219		32.4349		32.3910
7-Iso	50	<b>32.4826</b>	32.4851	<b>32.4053</b>	32.4152	<b>32.3902</b>	32.3798
8-Iso	70		32.4435		32.3934		32.3678
9-Iso	120		32.4007		32.3713		32.3562
10-Iso	150		32.3888		32.3653		32.3531

Table 5.23 Bending moment  $M_{34}$  vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

The above data is for Graphs 5.46, 5.47 & 5.48. The highlighted values are the verification of results.

Force-displacement relations in soil are implemented through the stiffness method to investigate the soil-structure interaction. In this way the soil response is contributed in the system analysis. This well-known approach is frequently utilised by many researchers (e. g. Lee 1975, Melerski 1992).

The verification tables can be obtained in available in Appendix A5-4 (Tables A5.46 – A5.52).

## **5.8. CONCLUDING REMARKS TO CHAPTER**

In this chapter, a 2D-hinged-support frame, subjected to a set of loads and founded on an elastic medium, was analysed using three different models (Winkler, isotropic and cross-anisotropic homogeneous elastic half-space). The elastic properties of the soil medium were provided by elastic parameters (Winkler spring coefficient, moduli of elasticity and Poisson's ratios). *SASIAP* software package was used to analyse this frame. The application of force-displacement relations was verified for a frame with full restrained-support using Space Gass software and applying the support displacements obtained from *SASIAP* as prescribed displacement loads. The results of system analysis were tabulated for internal force quantities (element end-moments) and nodal displacements. Results for the Winkler model based on isotropic elastic half-space values were also tabulated. The outputs are discussed in detail in chapter 6.

# CHAPTER

# SIX

*DISCUSSION OF RESULTS*

*Elastostatic Interaction Analysis of Frames  
Resting on Homogeneous Elastic Half-space*

## CHAPTER 6

### DISCUSSION OF RESULTS

#### 6.1. INTRODUCTION

In Chapter 5 the outputs of structure (example) analyses (using the software described in Chapter 4 by incorporating typical soil parameters for a typical plane frame structure) are presented. The structure was pin-connected to isolated foundations and founded on isotropic and cross-anisotropic media.

This chapter reviews the performance of the applied technique by evaluating the effects of variations of soil parameters associated with the used soil models. The outputs of the structural analysis (bending moments, nodal displacements and rotations) obtained in Chapter 5 are discussed in detail.

A review of results with particular emphasis on development and variations of bending moments and displacements in the typical plane frame, due to the variations in the soil parameters, reveals that the bending moments and the nodal displacements are more sensitive to variation in the modulus of elasticity ( $E_s$ ) than to variation in the soil Poisson's ratio ( $\nu$ ).

#### 6.2. OBSERVATIONS AND DISCUSSIONS ON RESULTS OF ANALYSING FRAME ON ISOTROPIC ELASTIC HALF-SPACE

The author has considered a number of examples with a typical range of elastic parameters for the soil and the main observations of the results are discussed in sections 6.2.1 to 6.2.3.

In Tables 5.1 & 5.1a, based on soil parameters used in cases 1 and 2, (Gerrard 1967) two isotropic soils ( $\nu = 0.25$  and  $\nu = 0.43$ ) are considered.

### 6.2.1. Case One of an Isotropic Soil

In isotropic soil case one, the output graphs (Graphs 5.1 - 5.9), a general hyperbolic trend is repeatedly observed in all the obtained curves for  $D$  (isolated footing interaction),  $P$  (plane interaction) and  $S$  (spatial interaction). These curves show an asymptotic trend in both horizontal and vertical axes. In the bending moment or nodal displacement versus the soil stiffness curves (Graphs 5.1 to 5.3), with a reduction in stiffness, there are increases in internal forces or nodal displacements. The percentages associated with such increases are greater for the bending moments than the nodal displacements.

In the bending moment and nodal displacement graphs, the curve  $P$  (associated with footing interactions within the frame plane) is located between curves  $D$  (the isolated footing interaction) and  $S$  (spatial interaction). This situation for such a given frame occurs as the accumulation of interaction received from adjacent supports (within the frame and neighbouring frames) exceeds the interaction at isolated footings. However, if the distances between the supports within the frame and the neighbouring frames vary these curves ( $D$ ,  $P$  and  $S$ ) may have different positions with respect to each other — which can be investigated by conducting a parametric study.

Tables 6.1 to 6.9 summarise the general response of a typical plane frame with isolated footings on isotropic elastic half-space. The bending moments for the case of  $\nu = 0.25$  associated with three cases of interaction (isolated interaction  $D$ ; plane frame interaction  $P$ ; spatial interaction  $S$ ) are drawn in Graphs 5.1 to 5.9.

**Variation of bending moments  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  associated with Graphs 5.1 to 5.9 for  $\nu=0.25$ ; and Graphs 5.10 to 5.18 for  $\nu=0.43$  respectively**

Model	Sub-case	Soil type	$M_{21}$	%	$M_{32}$	%	$M_{34}$	%
D (isolated footing- 1D)	1, $\nu=0.25$	Isotropic	21.400	-	-30.477	-	32.710	-
	2, $\nu=0.43$	Isotropic	22.090	3.2	-30.972	1.6	32.356	1.1
P (Plane frame- 2D)	1, $\nu=0.25$	Isotropic	21.400	-	-30.477	-	32.540	-
	2, $\nu=0.43$	Isotropic	22.245	3.8	-30.994	1.7	32.153	1.2
S (Spatial frame- 3D)	1, $\nu=0.25$	Isotropic	20.768	-	-29.968	-	32.460	-
	2, $\nu=0.43$	Isotropic	21.895	5.4	-30.600	2.1	32.085	1.2

Table 6.1 Variation of bending moment  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  for  $E_s= 20$  MPa, corresponding items ( $\nu_{0.25}-\nu_{0.43}$ ) in percentage. Base values are  $\nu=0.25$  for individual cases D, P & S.

Model	Sub-case	Soil type	$M_{21}$	%	$M_{32}$	%	$M_{34}$	%
D (isolated footing- 1D)	1, $\nu=0.25$	Isotropic	23.778	11.0	-33.908	11.3	32.480	0.7
	2, $\nu=0.43$	Isotropic	24.080	9.1	-34.154	10.3	32.342	0.4
P (Plane frame- 2D)	1, $\nu=0.25$	Isotropic	23.778	13.1	-33.908	11.3	32.420	0.3
	2, $\nu=0.43$	Isotropic	24.153	8.6	-34.176	10.3	32.260	0.3
S (Spatial frame- 3D)	1, $\nu=0.25$	Isotropic	23.480	13.0	-33.664	12.3	32.380	0.2
	2, $\nu=0.43$	Isotropic	23.995	9.6	-33.994	12.7	32.232	0.5

Table 6.2 Variation of bending moment  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  for  $E_s= 50$  MPa, corresponding items (Table 6.2 - Table 6.1) in percentage. Table 6.1 is the base line.

Model	Sub-case	Soil type	$M_{21}$	%	$M_{32}$	%	$M_{34}$	%
D (isolated footing- 1D)	1, $\nu=0.25$	Isotropic	24.271	13.4	-34.628	13.6	32.444	0.8
	2, $\nu=0.43$	Isotropic	24.496	10.8	-34.843	12.5	32.341	0.5
P (Plane frame- 2D)	1, $\nu=0.25$	Isotropic	24.271	13.4	-34.642	13.7	32.393	0.4
	2, $\nu=0.43$	Isotropic	24.554	10.4	-34.843	12.4	32.283	0.4
S (Spatial frame- 3D)	1, $\nu=0.25$	Isotropic	24.063	15.9	-34.462	15.0	32.368	0.2
	2, $\nu=0.43$	Isotropic	24.438	11.6	-34.710	13.4	32.262	0.6

Table 6.3 Variation of bending moment  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  for  $E_s= 70$  Mpa, corresponding items (Table 6.3 - Table 6.1) in percentage. Table 6.1 is the base line.



Variation of nodal displacements/rotation  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$  associated with  
 Graphs 5.1 to 5.9 for  $\nu=0.25$ ; and Graphs 5.10 to 5.18 for  $\nu=0.43$  respectively

Model		Sub-case	Soil type	$\Delta_{x1}$	%	$\Delta_{y1}$	%	$\Phi_{z1}$	%
D (isolated)	footing- 1D)	1, $\nu=0.25$	Isotropic	0.00070	-	-0.00067	-	-0.00893	-
		2, $\nu=0.43$	Isotropic	0.00020	71.0	-0.00060	10.5	-0.00893	0.3
P (Plane)	frame- 2D)	1, $\nu=0.25$	Isotropic	0.00090	-	-0.00097	-	-0.00891	-
		2, $\nu=0.43$	Isotropic	0.00040	56.0	-0.00080	18.0	-0.00892	0.1
S (Spatial)	frame- 3D)	1, $\nu=0.25$	Isotropic	0.00180	-	-0.00184	-	-0.00888	-
		2, $\nu=0.43$	Isotropic	0.00120	33.0	-0.00160	13.0	-0.00892	0.4

Table 6.4 Nodal displacements/rotation variation  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$  for  $E_s= 20$  MPa, corresponding items ( $\nu_{0.25} - \nu_{0.43}$ ) in percentage. Base values are  $\nu=0.25$  for individual cases D, P and S.

Model		Sub-case	Soil type	$\Delta_{x1}$	%	$\Delta_{y1}$	%	$\Phi_{z1}$	%
D (isolated)	footing- 1D)	1, $\nu=0.25$	Isotropic	0.00030	57.0	-0.00023	65.0	-0.00871	2.0
		2, $\nu=0.43$	Isotropic	0.00010	50.0	-0.00020	67.0	-0.00870	2.0
P (Plane)	frame- 2D)	1, $\nu=0.25$	Isotropic	0.00040	56.0	-0.00035	64.0	0.00870	2.3
		2, $\nu=0.43$	Isotropic	0.00020	50.0	-0.00030	62.0	-0.00870	2.2
S (Spatial)	frame- 3D)	1, $\nu=0.25$	Isotropic	0.00070	61.0	-0.00069	62.0	-0.00868	2.2
		2, $\nu=0.43$	Isotropic	0.00050	58.0	-0.00060	62.0	-0.00870	2.4

Table 6.5 Nodal displacements/rotation variation  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$  for  $E_s= 50$  Mpa corresponding items (Table 6.5 - Table 6.4) in percentage. Table 6.4 is the base line.

Model		Sub-case	Soil type	$\Delta_{x1}$	%	$\Delta_{y1}$	%	$\Phi_{z1}$	%
D (isolated)	footing- 1D)	1, $\nu=0.25$	Isotropic	0.00020	71.0	-0.00016	76.0	-0.00866	2.7
		2, $\nu=0.43$	Isotropic	0.00010	50.0	-0.00010	83.0	-0.00866	3.0
P (Plane)	frame- 2D)	1, $\nu=0.25$	Isotropic	0.00030	67.0	-0.00024	75.0	-0.00866	2.8
		2, $\nu=0.43$	Isotropic	0.00010	75.0	-0.00020	75.0	-0.00866	2.8
S (Spatial)	frame- 3D)	1, $\nu=0.25$	Isotropic	0.00050	72.0	-0.00049	73.0	-0.00865	2.0
		2, $\nu=0.43$	Isotropic	0.00040	67.0	-0.00040	75.0	-0.00866	2.8

Table 6.6 Nodal displacements/rotation variation  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$  for  $E_s= 70$  MPa corresponding items (Table 6.6 - Table 6.4 ) in percentage. Table 6.4 is the base line.

**Variation of nodal displacements/rotation  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  associated with  
Graphs 5.1 to 5.9 for  $\nu=0.25$ ; and Graphs 5.10 to 5.18 for  $\nu=0.43$  respectively**

Model		Sub-case	Soil type	$\Delta_{x3}$	%	$\Delta_{y3}$	%	$\Phi_{z3}$	%
D (isolated)	footing- 1D)	1, $\nu=0.25$	Isotropic	0.01529	-	-0.01451	-	-0.00337	-
		2, $\nu=0.43$	Isotropic	0.01481	3.1	-0.01399	3.6	-0.00329	12.1
P (Plane)	frame- 2D)	1, $\nu=0.25$	Isotropic	0.01543	-	-0.01470	-	-0.00335	-
		2, $\nu=0.43$	Isotropic	0.01495	3.1	-0.01416	3.6	-0.00326	2.7
S (Spatial)	frame- 3D)	1, $\nu=0.25$	Isotropic	0.01627	-	-0.01578	-	-0.00337	-
		2, $\nu=0.43$	Isotropic	0.01578	3.0	-0.01511	4.2	-0.00328	2.7

Table 6.7 Nodal displacements/rotation variation  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  for  $E_s= 20$  MPa, corresponding items ( $\nu_{0.25} - \nu_{0.43}$ ) in percentage. Base values are  $\nu=0.25$  for individual cases D, P and S.

Model		Sub-case	Soil type	$\Delta_{x3}$	%	$\Delta_{y3}$	%	$\Phi_{z3}$	%
D (isolated)	footing- 1D)	1, $\nu=0.25$	Isotropic	0.01469	3.9	-0.01205	17.0	-0.00307	8.9
		2, $\nu=0.43$	Isotropic	0.01400	5.5	-0.01184	15.0	-0.00304	7.6
P (Plane)	frame- 2D)	1, $\nu=0.25$	Isotropic	0.01478	4.2	-0.01213	17.5	-0.00306	8.9
		2, $\nu=0.43$	Isotropic	0.01406	6.0	-0.01190	16.0	-0.00303	7.0
S (Spatial)	frame- 3D)	1, $\nu=0.25$	Isotropic	0.01534	5.7	-0.01257	20.0	-0.00307	8.9
		2, $\nu=0.43$	Isotropic	0.01439	8.8	-0.01229	18.7	0.00304	7.3

Table 6.8 Nodal displacements/rotation variation  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  for  $E_s= 50$  MPa corresponding items (Table 6.8 - Table 6.7) in percentage. Table 6.7 is the base line.

Model		Sub-case	Soil type	$\Delta_{x3}$	%	$\Delta_{y3}$	%	$\Phi_{z3}$	%
D (isolated)	footing- 1D)	1, $\nu=0.25$	Isotropic	0.01420	7.1	-0.01157	20.0	0.00301	10.7
		2, $\nu=0.43$	Isotropic	0.01384	6.5	-0.01141	18.4	-0.00299	9.1
P (Plane)	frame- 2D)	1, $\nu=0.25$	Isotropic	0.01425	7.6	-0.01162	21.0	-0.00301	10.0
		2, $\nu=0.43$	Isotropic	0.01388	7.2	-0.01141	19.4	0.00298	8.6
S (Spatial)	frame- 3D)	1, $\nu=0.25$	Isotropic	0.01459	10.3	-0.01194	24.0	-0.00301	10.7
		2, $\nu=0.43$	Isotropic	0.01412	10.5	-0.01174	22.0	-0.00299	8.8

Table 6.9 Nodal displacements/rotation variation  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  for  $E_s= 70$  MPa corresponding items (Table 6.9 - Table 6.7) in percentage. Table 6.7 is the base line.

### 6.2.2. Case Two of an Isotropic Soil

Similar to case one of isotropic soil (section 6.2.1), in this soil case, the output graphs (Graphs 5.10 - 5.18) show a general hyperbolic trend which is repeatedly observed in all the curves  $D$ ,  $P$  and  $S$ . Bending moment and nodal displacement versus the soil stiffness curves show large internal forces and nodal displacements variations for softer soil. Moreover, these variations are greater in bending moments in comparison to nodal displacements. For instance, bending moment  $M_{21}$  associated with curve  $D$  varies 11 and 13.4% (see Tables 6.1 to 6.3) for soil stiffness  $E_s = 20, 50$  and  $70$  MPa respectively, and the corresponding variations for nodal displacements  $\Delta_{x3}$  are 3.9 and 7.1% (see Tables 6.7 to 6.9).

As expected, it is seen that nodal displacement becomes negligible as soil stiffness increases. Variation in bending moment as a function of soil stiffness also becomes negligible when soil stiffness exceeds around 30 to 50 MPa.

Similarly to the case in section 6.2.1, in Graphs 5.10 to 5.18 associated with the bending moments and nodal displacements, the curve  $P$  is located between curves  $D$  and  $S$ . The soil-structure interaction received from the individual footings is relatively small (due to relatively large bearing areas) in comparison to the interaction from cases  $P$  and  $S$  (as the distances between these footings and the adjacent frames are not significantly large, which cause higher interaction on the structure). In comparing Graphs 5.3 with 5.12, the bending moment  $M_{34}$  versus the soil modulus of elasticity, it can be seen that in Graph 5.12, curve  $D$  follows the same trend as Graph 5.3 does. That is the bending moment decreases when the soil stiffness increases while curves  $P$  and  $S$  do otherwise. This behaviour may help estimation of internal forces when considering three-dimensional frames.

Similarly, for the case of  $\nu = 0.43$  the structure response fields are shown in Graphs 5.10 to 5.18. Tables 6.1 to 6.9 are arranged for three different values of soil modulus of elasticity ( $E_s = 20, 50$  and  $70$  MPa) and the quantity changes were calculated and are shown as percentage values for easy comparison.

### 6.2.3. Comparison of Cases One and Two of the Isotropic Soil

Tables 6.1 to 6.9 are prepared for the purpose of comparing structural responses associated with two different isotropic soils ( $\nu = 0.43$  and  $\nu = 0.25$ ). Tables 6.1, 6.4 and 6.7 show the percentage changes of output fields for  $\nu = 0.43$  as compared with the corresponding quantities for  $\nu = 0.25$  in the same table used as the base case. However, in Tables 6.2, 6.3, 6.5, 6.6, 6.8 and 6.9, the percentage values are calculated based on the comparison with the corresponding values in Tables 6.1, 6.4 and 6.7 (taken as base cases) accordingly.

Tables 6.1 to 6.3 show that for a given soil Poisson's ratio ( $\nu = 0.25$ ), the bending moments  $M_{21}$  and  $M_{32}$  have a direct relationship with the soil Young's modulus ( $E_s = 20$  to  $70$  MPa). That is increase in bending moment due to the increase in soil modulus of elasticity can be observed. There is also an increase in bending moment due to the increase in soil Poisson's ratio. When  $\nu$  increases from  $0.25$  to  $0.43$ , bending moment  $M_{21}$  increases from  $21.4$  kN.m to  $22.09$  for case  $D$ , and from  $20.768$  kN.m to  $21.896$  kN.m in case  $S$ . Similar variations are observed for  $M_{32}$ . Table 6.2 shows that the bending moments for  $\nu = 0.25$  for three cases of  $D$ ,  $P$  and  $S$  interactions possess the highest variations.

In Tables 6.4 to 6.9, the nodal displacements and rotations for nodes 1 and 3 are presented. Tables 6.5 and 6.6 show that the variations of horizontal and vertical nodal displacements ( $\Delta x_1$ ,  $\Delta y_1$ ,  $\Delta x_3$ , and  $\Delta y_3$ ) are quite large (50-83%) in comparison to the nodal rotation variations  $\Phi_{z1}$  and  $\Phi_{z2}$  (2-3%). On the other hand, the corresponding variation quantities shown in Tables 6.8 and 6.9 are 3-24% for the nodal displacements and 7-10.5% for the nodal rotations respectively.

Graphs 5.1 to 5.18 show that the structural response fields represented by curves  $P$  (interaction confined to plane frame) are relatively close to those represented by curves  $S$  (spatial interaction). However, there is a significant difference between curves  $P$  and  $D$ . In general, one should expect that when the frame bay length increases the difference between the corresponding curves  $P$  and  $D$  diminishes.

The curves for the estimation of displacements (Graphs 5.1 to 5.18), although having the form of the elastic-type solutions, are in fact non-linear functions of the soil parameters. The influence of certain basic factors on the deformations of cohesionless soils, such as, the effect of stress history and the dimensions of the foundation are considerable (Papadopoulos, 1992). Therefore, it is important to have accurate soil data for evaluating displacements of shallow foundations. This problem is especially important when the foundation rests on a cohesionless soil, owing to the impossibility of having undisturbed samples (Maugen et al. 1998). The author has restricted the study to use of data presented in previous studies (Gerrard 1976 and Wardle 1977) in soil-structure analysis.

Although the discussion in this work is confined to soil-structure interaction on the surface of the soil mass, a similar approach can be used to analyse frames with supports within the elastic medium. In this case, the appropriate solutions such as Mindlin's relationships for loads and deformations should be used instead of those applied in this work.

### **6.3. OBSERVATIONS AND DISCUSSIONS ON RESULTS OF ANALYSING FRAME ON CROSS-ANISOTROPIC ELASTIC HALF-SPACE**

Tables 6.10 to 6.15 summarise the response of a plane frame (Graphs 5.19 to 5.36) with isolated footings on a soil medium, which is modelled as a cross-anisotropic elastic half-space. The five independent parameters defining the cross-anisotropic medium used in the analysis have been restricted to typical values obtained from earlier studies. Similarly to Table 6.3, the variations in internal forces (the bending moments) are shown in Table 6.11.

The range of soil parameters considered for analysis in cross-anisotropic case was not wide enough to show a full spectrum of the outputs. However, even with the existing limited input, curves in Graphs 5.19 to 5.36 show already general trends of the solutions.

Graphs 5.19 to 5.36 show that the structural response fields represented by curves  $P$  (interaction confined to plane frame) are relatively close to those represented by

curves  $D$  (isolated footing interaction). However, there is a significant difference between curves  $P$  and  $S$ . An explanation for this behaviour, which is in contrast with isotropic cases in section 6.2, is that the internal force redistribution through the frame itself has stronger effect on the analysis fields than from the adjacent frames. This may be due to the variation of cross-anisotropic soil properties in different directions whereas in isotropic soil there is a uniformity of properties in all directions. Consequently, the increases bay dimension is expected to result in increases the difference between the corresponding curves  $P$  and  $D$  as the interaction via soil medium weakens.

In Graphs 5.20, 5.21, 5.22 and 5.25 associated with  $M_{32}$ ,  $M_{34}$ ,  $\Delta_{x1}$ , and  $\Delta_{x3}$  respectively, it is observed that these fields variations have inverse relationships with  $E_v$  in a cross-anisotropic soil. While in Graph 5.21, the bending moments  $M_{34}$  associated with isolated footings ( $D$ ) are higher than the corresponding quantities for plane ( $P$ ) and spatial ( $S$ ) interactions, the curves  $S$  in the other three graphs present higher values as compared with the corresponding values in curves  $D$  and  $P$ . This trend indicates that the isolated footing case ( $D$ ) has a stronger effect on the internal force  $M_{34}$  than the cases  $P$  and  $S$ . However, as shown in Table 6.10, in the plane frame interaction (curve  $P$ ), due to contribution of interaction from nodes within the frame despite the stress redistribution, the interaction effect remains relatively large. Although, such variations are moderate in a softer soil.

**Variation of bending moments  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  associated with Graphs 5.19 to 5.36 for  $E_v=26$  and 35 MPa,  $\nu_{HV}=0.30$  ; (refer to Table 5.1b)**

Model	Sub-case, $\nu_{HV}$ $E_v$ (MPa)	Soil type	$M_{21}$	%	$M_{32}$	%	$M_{34}$	%
D (isolated footing- 1D)	3, 0.30, 26	Cross-anisotropic	21.8510	-	-32.4010	-	33.3255	-
	3, 0.30, 35	Cross-anisotropic	22.7303	4.0	-33.3753	3.0	33.0790	0.7
P (Plane frame- 2D)	3, 0.30, 26	Cross-anisotropic	21.8785	-	-32.4440	-	33.2285	-
	3, 0.30, 35	Cross-anisotropic	22.7546	4.0	-33.4120	0.1	33.0057	0.7
S (Spatial frame- 3D)	3, 0.30, 26	Cross-anisotropic	21.4490	-	-32.6470	-	33.1741	-
	3, 0.30, 35	Cross-anisotropic	22.4214	4.5	-33.1142	1.6	32.9647	0.6

Table 6.10 Variation of bending moment  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  for  $E_v= 26$  MPa and  $E_v=35$  MPa in a soft to medium soil, corresponding ( $E_v 35 - E_v 26$ ) in percentage.

Case  $E_s=26$  MPa is the base line for individual cases D, P, S.

**Variation of bending moments  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  associated with Graphs 5.19 to 5.36 for  $E_v=75$  and 84 MPa,  $\nu_{HV}=0.35$ ; (refer to Table 5.1b)**

Model	Sub-case, $\nu_{HV}$ $E_v$ (MPa)	Soil type	$M_{21}$	%	$M_{32}$	%	$M_{34}$	%
D (isolated footing- 1D)	5, 0.35, 75	Cross-anisotropic	24.3209	11.3	-35.0783	8.3	32.6218	2.1
	5, 0.35, 84	Cross-anisotropic	24.4544	7.6	-35.2359	5.6	32.5920	1.5
P (Plane frame- 2D)	5, 0.35, 75	Cross-anisotropic	24.3358	11.2	-35.1026	8.2	32.5930	1.9
	5, 0.35, 84	Cross-anisotropic	24.4680	7.5	-35.2578	5.5	32.5662	1.3
S (Spatial frame- 3D)	5, 0.35, 75	Cross-anisotropic	24.1869	12.8	-34.9631	7.0	32.5762	1.8
	5, 0.35, 84	Cross-anisotropic	24.3341	8.5	-35.1323	6.0	32.5513	1.2

Table 6.11 Variation of bending moment  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  for  $E_v= 75$  MPa and  $E_v= 84$  MPa

in a medium to hard soil, corresponding (Table 6.11 - Table 6.10) in percentage

Table 6.10 is base line.

**Variation of nodal displacements/rotation  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$  associated with Graphs 5.19 to 5.36 for  $E_v=26$  and 35 MPa ,  $\nu_{HV}=0.30$ ; (refer to Table 5.1b)**

Model	Sub-case, $\nu_{HV}$ $E_v$ (MPa)	Soil type	$\Delta_{x1}$	%	$\Delta_{y1}$	%	$\Phi_{z1}$	%
D (isolated footing- 1D)	3, 0.30, 26	Cross-anisotropic	0.00140	-	-0.00045	-	-0.00872	-
	3, 0.30, 35	Cross-anisotropic	0.00105	25.0	-0.00020	56.0	-0.00868	0.4
P (Plane frame- 2D)	3, 0.30, 26	Cross-anisotropic	0.00153	-	-0.00067	-	-0.00871	-
	3, 0.30, 35	Cross-anisotropic	0.00115	24.8	-0.00048	28.4	-0.00867	0.4
S (Spatial frame- 3D)	3, 0.30, 26	Cross-anisotropic	0.00208	-	-0.00130	-	-0.00872	-
	3, 0.30, 35	Cross-anisotropic	0.00156	25.0	-0.00094	27.7	-0.00865	0.8

Table 6.12 Nodal displacements/rotation variation  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$  for  $E_v= 26$  MPa and

35 MPa in a soft to medium soil, corresponding ( $E_v 35 - E_v 26$ ) in percentage

Case  $E_s=26$  MPa is the base line for individual cases D, P, S.

**Variation of nodal displacements/rotation  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$  associated with Graphs 5.19 to 5.36 for  $E_v=75$  and 84 MPa ,  $\nu_{HV}=0.35$ ; (refer to Table 5.1b)**

Model	Sub-case, $\nu_{HV}$ $E_v$ (MPa)	Soil type	$\Delta_{x1}$	%	$\Delta_{y1}$	%	$\Phi_{z1}$	%
D (isolated footing- 1D)	5, 0.35, 75	Cross-anisotropic	0.00041	70.7	-0.00012	73.3	-0.00860	1.4
	5, 0.35, 84	Cross-anisotropic	0.00037	65.0	-0.00011	45.0	-0.00861	0.8
P (Plane frame- 2D)	5, 0.35, 75	Cross-anisotropic	0.00045	70.0	-0.00020	70.0	-0.00860	1.3
	5, 0.35, 84	Cross-anisotropic	0.00040	65.0	-0.00018	62.0	-0.00860	0.8
S (Spatial frame- 3D)	5, 0.35, 75	Cross-anisotropic	0.00061	71.0	-0.00040	69.0	-0.00860	1.4
	5, 0.35, 84	Cross-anisotropic	0.00055	65.0	-0.00036	62.0	-0.00860	0.6

Table 6.13 Nodal displacements/rotation variation  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$  for  $E_v= 75$  MPa and

84 MPa in a medium to hard soil, corresponding (Table 6.13 - Table 6.12) in percentage

Table 6.12 is base line.



**Variation of nodal displacements/rotation  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  associated with Graphs 5.19 to 5.36 for  $E_v=26$  and 35 MPa ,  $\nu_{HV}=0.30$  ,(Table 5.1b)**

Model	Sub-case, $\nu_{HV}$ $E_v$ (MPa)	Soil type	$\Delta_{x3}$	%	$\Delta_{y3}$	%	$\Phi_{z3}$	%
D (isolated footing- 1D)	3, 0.30, 26	Cross-anisotropic	0.01539	-	-0.01335	-	-0.00327	-
	3, 0.30, 35	Cross-anisotropic	0.01490	3.2	-0.01260	5.6	0.00317	3.0
P (Plane frame- 2D)	3, 0.30, 26	Cross-anisotropic	0.01547	-	-0.01349	-	-0.00326	-
	3, 0.30, 35	Cross-anisotropic	0.01495	3.3	-0.01270	5.8	-0.00316	3.1
S (Spatial frame- 3D)	3, 0.30, 26	Cross-anisotropic	0.01595	-	-0.01426	-	-0.00327	-
	3, 0.30, 35	Cross-anisotropic	0.01534	3.8	-0.01328	7.2	-0.00317	3.0

Table 6.14 Nodal displacements/rotation variation  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  for  $E_v= 26$  MPa and  $E_v=35$  MPa in a soft to medium soil, corresponding ( $E_{v35} - E_{v26}$ ) in percentage  
Case  $E_s=26$  MPa is the base line for individual cases D, P, S.

**Variation of nodal displacements/rotation  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  associated with Graphs 5.19 to 5.36 for  $E_v=75$  and 84 MPa ,  $\nu_{HV}=0.35$  ,(Table 5.1b)**

Model	Sub-case, $\nu_{HV}$ $E_v$ (MPa)	Soil type	$\Delta_{x3}$	%	$\Delta_{y3}$	%	$\Phi_{z3}$	%
D (isolated footing- 1D)	5, 0.35, 75	Cross-anisotropic	0.01405	8.7	-0.01135	15.0	-0.00300	8.2
	5, 0.35, 84	Cross-anisotropic	0.01398	6.2	-0.01124	10.8	-0.00300	5.4
P (Plane frame- 2D)	5, 0.35, 75	Cross-anisotropic	0.01407	9.0	-0.01139	15.6	-0.00300	8.0
	5, 0.35, 84	Cross-anisotropic	0.01400	6.4	-0.01128	11.2	-0.00300	5.0
S (Spatial frame- 3D)	5, 0.35, 75	Cross-anisotropic	0.01422	10.8	-0.01165	18.3	-0.00300	8.2
	5, 0.35, 84	Cross-anisotropic	-0.01414	7.8	-0.01151	13.3	-0.00300	5.3

Table 6.15 Nodal displacements/rotation variation  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  for  $E_v= 75$  MPa and  $E_v= 84$  MPa in a medium to hard soil,corresponding (Table 6.15 -Table 6.14) in percentage  
Table 6.14 is base line.

Graphs 5.37 to 5.45 are obtained based on cross-anisotropic soil parameters. In this set, one of the independent soil parameters (soil shear modulus of elasticity in vertical direction) is varied to investigate its effect on the frame internal forces and nodal displacements. Tables 6.16 to 6.18 provide numerical summary of the outputs.

In the cross-anisotropic soil in this study, with interim parameters, when  $\beta^2$  takes quantities such as positive, zero and negative in conjunction with  $\alpha^2$ , the bending moments  $M_{21}$  and  $M_{32}$ , and the nodal displacements  $\Delta_{x3}$  and  $\Delta_{y3}$  show a greater increase (up to 3.2 %) than bending moment  $M_{34}$  and nodal rotations  $\Phi_{z1}$  and  $\Phi_{z3}$  (see Tables 6.16 to 6.18). However, the nodal displacements  $\Delta_{x1}$  and  $\Delta_{y1}$  show no significant response (almost zero %) to the changes in the soil parameter. In comparing with the isotropic soil, the case of cross-anisotropic soil bed is associated with larger variations in the nodal displacements.

**Variation of bending moments  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  and nodal displacements/rotations  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$ ,  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  associated with Graphs 5.37 to 5.45**

Model	Sub-case	Soil type	$M_{21}$ & %		$M_{32}$ & %		$M_{34}$ & %	
D (isolated)	footing- 1D)	10, Fv=21.429MPa	Special Cross-anisotropic ( $\alpha^2=1$ ; $\beta^2=0$ )		22.129   -		-33.229   -	
		7, Fv=21.450MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 > 0$ )		22.370   1.1		-33.629   1.2	
		9, Fv=22.627MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 = 0$ )		22.439   1.4		-33.689   1.4	
		8, Fv=23.000MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 < 0$ )		22.460   1.5		-33.707   1.4	
P (Plane)	frame- 2D)	10, Fv=21.429MPa	Special Cross-anisotropic ( $\alpha^2=1$ ; $\beta^2=0$ )		22.218   -		-33.258   -	
		7, Fv=21.450MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 > 0$ )		22.466   1.1		-33.652   1.2	
		9, Fv=22.627MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 = 0$ )		22.532   1.4		-33.712   1.4	
		8, Fv=23.000MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 < 0$ )		22.561   1.5		-33.733   1.4	
S (Spatial)	frame- 3D)	10, Fv=21.429MPa	Special Cross-anisotropic ( $\alpha^2=1$ ; $\beta^2=0$ )		21.941   -		-32.958   -	
		7, Fv=21.450MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 > 0$ )		22.232   1.3		-33.384   1.3	
		9, Fv=22.627MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 = 0$ )		22.301   1.6		-33.449   1.5	
		8, Fv=23.000MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 < 0$ )		22.342   1.8		-33.476   1.6	

Table 6.16 Variation of bending moment  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  versus Soil shear modulus of elasticity Fv

Model	Sub-case	Soil type	$\Delta_{x1}$ & %		$\Delta_{y1}$ & %		$\Phi_{z1}$ & %	
D (isolated)	footing- 1D)	10, Fv=21.429MPa	Special Cross-anisotropic ( $\alpha^2=1$ ; $\beta^2=0$ )		0.0016   NSV		-0.0004   NSV	
		7, Fv=21.450MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 > 0$ )		0.0016   NSV		-0.0003   NSV	
		9, Fv=22.627MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 = 0$ )		0.0016   NSV		-0.0003   NSV	
		8, Fv=23.000MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 < 0$ )		0.0016   NSV		-0.0003   NSV	
P (Plane)	frame- 2D)	10, Fv=21.429MPa	Special Cross-anisotropic ( $\alpha^2=1$ ; $\beta^2=0$ )		0.0017   NSV		-0.0006   NSV	
		7, Fv=21.450MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 > 0$ )		0.0017   NSV		-0.0005   NSV	
		9, Fv=22.627MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 = 0$ )		0.0017   NSV		-0.0005   NSV	
		8, Fv=23.000MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 < 0$ )		0.0017   NSV		-0.0005   NSV	
S (Spatial)	frame- 3D)	10, Fv=21.429MPa	Special Cross-anisotropic ( $\alpha^2=1$ ; $\beta^2=0$ )		0.0023   -		-0.0011   NSV	
		7, Fv=21.450MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 > 0$ )		0.0023   0		-0.0010   NSV	
		9, Fv=22.627MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 = 0$ )		0.0022   4		-0.0009   NSV	
		8, Fv=23.000MPa	Cross-anisotropic ( $\alpha^2 < 1$ ; $\beta^2 < 0$ )		0.0022   4		-0.0009   NSV	

Table 6.17 Variation of nodal displacements/rotation  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$  versus Soil shear modulus of elasticity Fv

NB: NSV is No Significant Variation

Table 6.16 is base line for the comparison used for individual items in this table. Case  $\alpha^2=1$   $\beta^2=0$  is base line for the comparison between other three cases.

Variation of bending moments  $M_{21}$ ,  $M_{32}$ ,  $M_{34}$  and nodal displacements/rotations  $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$ ,  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  associated with Graphs 5.37 to 5.45

Model	Sub-case	Soil type	$\Delta_{x3}$ & %		$\Delta_{y3}$ & %		$\Phi_{z3}$ & %	
D (isolated footing- 1D)	10, Fv=21.429MPa	Special Cross-anisotropic ( $\alpha^2=1$ ; $\beta^2=0$ )	0.01537	-	-0.01285	-	-0.00323	-
	7, Fv=21.450MPa	Cross-anisotropic ( $\alpha^2<>1$ ; $\beta^2>0$ )	0.01527	0.6	-0.01258	0.0	-0.00320	0.9
	9, Fv=22.627MPa	Cross-anisotropic ( $\alpha^2<>1$ ; $\beta^2=0$ )	0.01530	0.4	-0.01253	2.5	-0.00319	1.2
	8, Fv=23.000MPa	Cross-anisotropic ( $\alpha^2<>1$ ; $\beta^2<0$ )	0.01522	1.0	-0.01252	2.6	-0.00319	1.2
P (Plane frame- 2D)	10, Fv=21.429MPa	Special Cross-anisotropic ( $\alpha^2=1$ ; $\beta^2=0$ )	0.01546	-	-0.01297	-	-0.00321	-
	7, Fv=21.450MPa	Cross-anisotropic ( $\alpha^2<>1$ ; $\beta^2>0$ )	0.01536	0.6	-0.01269	2.2	-0.00318	0.9
	9, Fv=22.627MPa	Cross-anisotropic ( $\alpha^2<>1$ ; $\beta^2=0$ )	0.01532	0.9	-0.01263	2.6	-0.00317	1.2
	8, Fv=23.000MPa	Cross-anisotropic ( $\alpha^2<>1$ ; $\beta^2<0$ )	0.01530	1.0	-0.01262	2.7	-0.00317	1.2
S (Spatial frame- 3D)	10, Fv=21.429MPa	Special Cross-anisotropic ( $\alpha^2=1$ ; $\beta^2=0$ )	0.01603	-	-0.01363	-	-0.00322	-
	7, Fv=21.450MPa	Cross-anisotropic ( $\alpha^2<>1$ ; $\beta^2>0$ )	0.01591	0.7	-0.01327	2.6	-0.00319	0.9
	9, Fv=22.627MPa	Cross-anisotropic ( $\alpha^2<>1$ ; $\beta^2=0$ )	0.01585	1.1	-0.01321	3.1	-0.00318	0.9
	8, Fv=23.000MPa	Cross-anisotropic ( $\alpha^2<>1$ ; $\beta^2<0$ )	0.01583	1.2	-0.01319	3.2	-0.00318	0.9

Table 6.18 Variation of nodal displacements/rotation  $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$  versus Soil shear modulus of elasticity Fv

Case  $\alpha^2=1$   $\beta^2=0$  is base line for the comparison between other three cases

It can be observed that element end bending moments in frames with isolated and plane interactions are very close (see Graphs 5.19, 5.20, 5.25, 5.26, 5.28). However, they differ quite significantly from the corresponding quantities associated with the space interactions.

According to the results from the cross-anisotropic cases (sub-cases 32 to 35), the soil elastic parameters are chosen such that these parameters make the two interim quantities  $\alpha$  and  $\beta$  (see sections 3.5.5, A2.1.2.1 and A2.2.2.2) become  $\alpha^2 = 1$  and  $\beta^2 = 0$ . In this group of examples, a especial response appears which is expected due to variations of  $\alpha^2$  and  $\beta^2$  when these two quantities together establish an isotropic state in the soil (Graphs 5.37 to 5.45). In Graphs 5.37, 5.38, 5.41 and 5.44 there is a rapid variation at the point representing isotropic status, while in the rest of graphs the transition status occurs smoother.

It can be noticed that for the cases considered the internal force variations (Tables 6.10 and 6.11) are much smaller (4 to 13 percent) than the variations of the support displacements (25 to 73 percent - Tables 6.12 and 6.13). Especially, the vertical displacements appear to be very sensitive to variations of soil properties. However, this feature of the soil-structure interaction should not be surprising since the internal force variations depend upon the differential settlements and not on the settlements themselves. Hence, the increase of settlements does not have to translate into increase of internal forces. It should be pointed out, however, that for cases of loads causing reactions of opposite sense at adjoining support the situation can be reversed. That means that relatively small changes in support displacements may be associated with relatively large variations of internal forces, because displacements at adjoining supports have opposite directions.

## 6.4. FINDINGS AND DISCUSSION ON RESULTS OF ANALYSIS: WINKLER MODEL AS COMPARED WITH ISOTROPIC ELASTIC HALF- SPACE

To investigate the dependency of structural response fields such as the bending moments, nodal displacements and rotations, on the variations of soil parameters, three sets of isotropic constants ( $E_s = 30$  MPa and  $\nu = 0.40, 0.15$  and  $0.49$ ) and the corresponding Winkler spring coefficients ( $k_s = 45.496, 39.096$  and  $50.292$  MN/m<sup>3</sup>) based on equal settlement of an isolated footing (Hemsley 1987) were considered. The Poisson's ratios were selected to represent three different cases: a mixed soil type, which is frequently observed, a soil with small volume changes and finally, a soil with large volume variation.

Tables 5.3 to 5.5 summarise the response (internal forces and nodal displacements) of a typical plane frame with isolated footings on a soil medium, which is modelled as being infinitely rigid. Similarly, for an isotropic soil medium and its corresponding Winkler's spring, Tables 5.6 to 5.14 are developed to review the interaction quantities.

Referring to Table 5.6, as the spring coefficient  $k_s$  increases, the bending moment  $M_{21}$  increases. For instance, for a 28.6% increase in the spring coefficient (from 39096 to 50292 kN/m<sup>3</sup>), the corresponding bending moment in isotropic model increased 4% (22.449 to 23.3858 kN.m) in isolated footing interaction case  $D$  and 6.6% increase (21.910 to 23.3635 kN/m<sup>3</sup>) in space frame interaction case  $S$ , while the variation in the bending moment due to Winkler is 30.5129 to 30.7013.

In Table 5.6, across from cases  $D$  to  $S$  in isotropic model, despite increases in the interaction between the isolated footings, there is reduction of 2.4% in bending moment (22.449 to 21.910 kN.m), which corresponds to Winkler spring coefficient of 39096 kN/m<sup>3</sup>, and a reduction of 0.1% in bending moments in isotropic model corresponding to the Winkler spring coefficient of 50292 kN/m<sup>3</sup>, respectively. Associating these cases with Table 5.8, the variations in the bending moment  $M_{34}$

are less than the previous case. In the case  $D$  in isotropic model, only 1.3% increase in this bending moment (32.6846 to 32.5186 kN.m) and only 1.4% (32.5186 to 32.0655 kN.m) in case  $S$  is observed, while the negative bending moment for the Winkler idealisation has less significant increase (in absolute value) 0.03% (-28.4515 to -28.4607 kN.m).

Displacements corresponding to support node (Node 1) in Tables 5.9 to 5.11, and for an inner structure node (Node 3) Tables 5.12 to 5.14 are shown respectively. Horizontal displacement  $\Delta_{x1}$  presents an increase of about 133% comparing cases  $D$  with  $S$  (0.0006 to 0.0014 m) in isotropic soil associated with a softer spring coefficient 39.096 MN/m<sup>3</sup>, while the maximum increase for such a displacement occurred in the stiffest soil (almost 0.0000 to 0.0006 m) in the case  $S$ , which is relatively significant.

For individual cases  $D$ ,  $P$  and  $S$  of the isotropic soil, the vertical deflection decreases a significant 25%, (0.0004 to 0.0003 m) corresponding with an increase in spring coefficient  $k_s$ . This situation is observed in the isotropic soil with the minimum variations from 0.0012 to 0.0010 m for the case  $S$ , and in the case  $D$  for the highest employed corresponding spring coefficient (50.292 MN/m<sup>3</sup>) there is no deflection at all (Tables 5.9 and 5.10). The variations in deflection imply that the isolated footings show strongest effect due to the spring stiffness  $k_s$  in compare to the isotropic cases of plane and spatial  $P$  and  $S$ .

Vertical displacements for an inner node (Node 3) shown in Table 5.13 indicate small decreases of 5%, (0.0133 to 0.0126 m in isotropic cases, and 0.0219 to 0.0215 m in Winkler model). The corresponding increase in the Winkler coefficient is from 39.096 to 50.292 MN/m<sup>3</sup>.

There are no significant changes in angles of rotation for Node 1 or Node 3 (Tables 5.11 and 5.14) associated with Winkler idealisation or cases of isotropic model at any interaction options ( $D$ ,  $P$  or  $S$ ).

Outcomes of the analysis from this research on the soil-structure interaction associated with the Winkler model in comparison to the isotropic idealisation are found similar to those of Lee and Brown (1972) regarding a continuous foundation beam. In comparison to the structure internal forces between the Winkler and the elastic half-space analyses, they assumed a correlation between the elastic soil parameters and Winkler springs based on equal settlements for an isolated footing. Lee and Brown (1972) found that the differences in maximum bending moment in the foundation beam for a typical frame structure were relatively small. However, there was an expected trend of decreasing maximum moment with increasing flexibility of the foundation beam.

## **6.5. SENSITIVITY ANALYSIS**

As described earlier in Chapter 2, determining soil parameters involves laboratory or in situ tests and associated measurements. These processes often incorporate errors that affect the accuracy of the soil-parameters obtained. To conduct a sensitivity analysis each parameter should be incremented or decremented for individual analysis and the interaction results assessed. Similarly to soil parameters, the elastic parameters of the superstructure also involve inaccuracies. As a result, the process of sensitivity of system analysis to parameters can be summarised as follows.

The overall soil-structure interaction analysis can be reduced to three hierarchical levels of parameters as illustrated in Figure 6.1 (Ang and Tang 1975 and 1984). The first level of variables considers the primary input variables (the environmental parameters such as moisture, consolidation, fissures and porosity) which are required as input into the models to determine elastic parameters such as Poisson's ratio and elastic moduli. The entire or some of the level one variables may depend upon the above-mentioned environmental parameters.



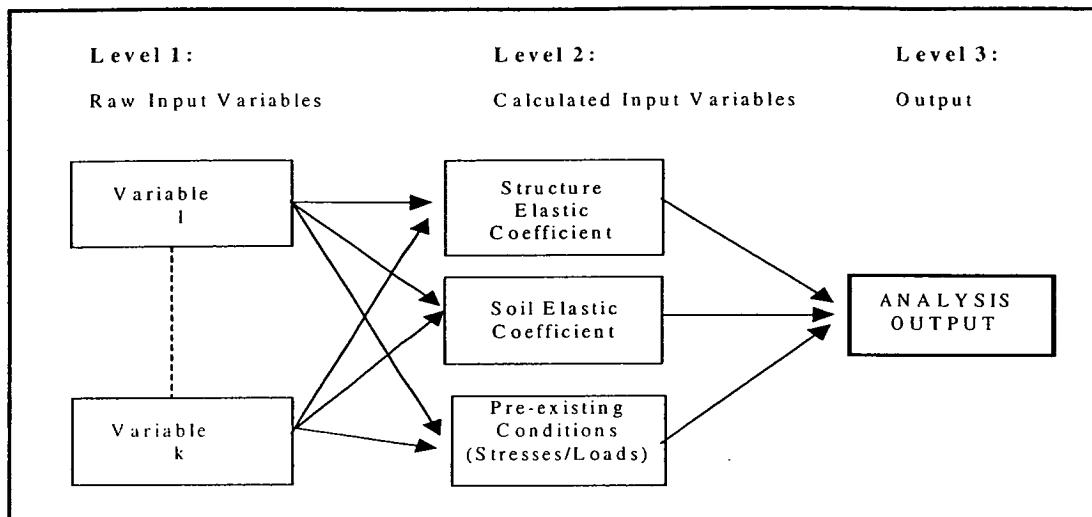


Figure 6.1 Illustration of parameters hierarchy in system analysis

The elastic coefficients (refer to the above figure) are considered as the level two parameters or intermediate variables. Finally, level three variables represent the final interaction analysis result.

The sensitivity of a given variable depends on the level at which it is being measured. For instance, a variable of level one, whilst having a significant influence on the next level variable, in the context of the overall results may in fact not be at all significant. This depends on how the soil model and the system analysis incorporate each variable and how the internal processes treat that variable.

Throughout the numerical examples of analysis conducted in this project, an overall view of influence of soil parameters is given in sections 6.2 to 6.4. However, for a broader scope of analysis, the problem should be viewed as involving random variables and, therefore, probability functions associated with each parameter should be used, which is beyond the scope of this project.

## 6.6. CONCLUDING REMARKS TO CHAPTER

The structure analysis output graphs, Graphs 5.1 to 5.18 associated with isotropic and Graphs 5.19 to 5.45 for cross-anisotropic elastic half-space are used to arrange Tables 6.1 to 6.9 and 6.10 to 6.18 respectively. These tables are prepared to classify the outputs and facilitate the conducted discussion. Bending moments  $M_{21}$ ,  $M_{32}$  and  $M_{34}$ ,

# CHAPTER

# SEVEN

*SUMMARY AND CONCLUSION*

*Elastostatic Interaction Analysis of Frames  
Resting on Homogeneous Elastic Half-space*

## CHAPTER 7

### SUMMARY AND CONCLUSIONS

#### 7.1. SUMMARY

The main objective of this project was the development of a technique of interaction analysis of soil-structure systems involving frames. In the structural modelling used, the classical direct stiffness method was employed whilst the stiffness matrix of the foundation bed was derived from the force-displacement relationships associated with the appropriate elastic half-space media representing the soil. Particular attention was given to planar frames, assuming that these are capable of adequately representing the actual, three-dimensional frame systems.

In the current design analysis of frames involving soil-structure interaction, Winkler foundation is the prevailing soil idealisation. Since Winkler foundation is widely regarded as a poor soil model, in this work, the concepts of homogeneous, elastic half-spaces with either isotropic or cross-anisotropic properties are employed to represent the media supporting the frames considered. As such media are known to be capable of providing better idealisations of soils than the Winkler foundation is, the proposed analysis of frames seems to possess a real, practical value.

The following three soil-structure interaction patterns are considered in this study:

- The interaction is confined to that of individual footings of the frame and the soil masses directly underneath those footings. That means that no interaction through the soil between the frame footings is taken into account. Because of the localised interaction pattern, the case is somewhat similar to that of Winkler springs. It is the simplest and the most crude soil-structure interaction model applied in this work.
- This soil-structure interaction pattern improves the above one by adding the interaction through the soil media among the footings of a plane frame.

- The third interaction model is somewhat more rigorous than the second one as it takes account of interaction among not only footings within a plane frame but also among the plane frames representing the actual, three-dimensional structure. This three-dimensional interaction, however, is only soil-induced as no structural connections between adjacent two-dimensional frames are considered.

## 7.2. CONCLUSIONS

This research shows that soils depending on their mechanical properties may considerably affect the responses of superstructures to applied loads. Therefore, the application of appropriate tools of structural analysis allowing for adequate modelling of soil-structure interaction is of genuine importance to design process. The technique of analysis proposed in this work can be viewed as right steps towards achieving the goal of better design of frame structures.

The examples of analysis provided in Chapter 5 and whose results were discussed in Chapter 6, although showing certain trends, aimed primarily at general illustration of effects of soil interaction on the frame structural responses. Since for a given frame, the structural response fields are non-linear functions of soil parameters linear interpolation of results is not recommended. However, as the proposed analysis is easy to perform it is no real problem to investigate the structural responses of frames for a relevant range of soil parameters. This would allow frame designers to base the structural dimensioning on the most unfavorable work conditions.

An alternative approach to safe design of frames related to soil-structure interactions is to put more effort into the process of evaluation of appropriate soil properties at the building site. However, in many situations this may be a difficult task. Cases of this nature will usually be associated with cohesionless soils where it is quite difficult to extract undisturbed soil samples. As a result, the properties of soil based on such samples exhibit poor correlation with the properties of the foundation bed. Also, in cross-anisotropic soils, the range of uncertainties is rather large making it difficult to

find appropriate correlation between the measured and real properties of the foundation bed.

The effects of interaction between the footings within the same frame and also between frames observed in the examples of Chapter 5 should not be taken as a guide to choose a particular interaction analysis variant ( $D$ ,  $P$  or  $S$ ). In general, the level of such interactions depends strongly not only on the soil parameters but also upon the distances between the footings in the actual three-dimensional structure. Consequently, in certain cases, it may be sufficient to use the simplest interaction variant (variant  $D$  - no soil-transmitted interaction between footings), in others, interaction between footings of a single, planar frame ( $P$ ) is needed, and yet in another cases, the space interaction ( $S$ ) is the appropriate choice. Also, it does not seem to be possible to develop a specific formula for this purpose.

One should not be surprised to see that the results of analysis associated with the Winkler idealisation of the soil do not correspond well to those from the analysis involving the elastic half-space foundation bed. The Winkler model is not equivalent to the elastic half-space model. Since Winkler springs have been found to be a worse representation of real soils than the elastic half-space media, it is important to have appropriate tools of analysis of frames employing such media.

The implementation of this study clearly requires very much less input than similar ones using the finite element method. As a result, the developed computer program could become an efficient and popular design tool.

While this study has not broken new ground, the approach developed by the author allows for relatively easy examination of sensitivity of frames to variations of soil parameters. This should contribute to better understanding of the soil-structure interaction as well as to enable the frame designers to better account for such an interaction. This work may also encourage other researchers to develop techniques and tools that will really break ground in this area.

### **7.3. FUTURE WORK**

This study, although relatively extensive, is by no means fully exhaustive. Consequently, further work in the area is recommended, which might proceed on a range of the associated topics. Some of these are listed below:

1. Extension of work to cover layered soil,
2. Development and implementation of computer procedures for analysis of frames with footings inside the soil mass. This should be relatively easy since appropriate force-displacement relations for such cases already exist (Mindlin's solutions 1936),
3. Extension of the present, deterministic approach to probabilistic elastic analysis that would better reflect the nature of soil properties,
4. A much more demanding extension of this work would be to consider non-linear behaviour of soil.

# REFERENCES

## REFERENCES

1. Abramowitz, M., and Stegun, A. (1970) *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover Publication Inc., New York, Edit 7.
2. The Asian Centre for Engineering Computations and Software News and Views (ACECOMS), *Computer Applications in Civil Engineering (1999)*, July-December 1999.
3. Ahlvin, R. G., and Ulery, H. H. (1962) Tabulated Values for Determining the Complete Pattern of Stresses, Strain, and Deflections Beneath a Uniform Load on a Homogeneous Half-space. *Highway Research Board*, Bulletin 342, pp. 1-13.
4. Akili, W. (1980) A Productive method for permanent deformation in bitumen emulsion treated sand, *Australian Road Research (ARRB)*, 10(3), pp. 3-11.
5. Ang, A. H. and Tang, W.H. (1975) *Probability Concepts in Engineering Planning and Designs: Volume I: Basic Principles*, John Wiley and Sons Inc, New York.
6. Ang, A. H. and Tang, W.H. (1984) *Probability Concepts in Engineering Planning and Designs: Volume II: Decision Risk and Reliability*, John Wiley and Sons Inc, New York.
7. Anon, (1960) De Spanningsverdeling in en Homogeen Anisotroop Elastic Half Medium. *L.G.M. Mededelingen*, Vol 5, No. 2, Delft Laboratory.
8. Antony, S.J., (1994) Interaction Analysis of Strip Footing Resting on a Non-homogeneous Elastic Half-plane. *NSSM-94, Natl. Symp. On Structural Mechanics*, Indian Institute of Science, Bangalore, pp. 152-157 (This author is also referred as Joseph Antony).
9. Antony, S.J., and Chandrashekhara K. (1997) Contact Stresses for a Footing on an Orthotropic Elastic Medium. *Strain*, November, pp. 127-132.
10. ASTM - American Society for Testing and Materials (1987) *ANSI/ASTM D1883-67. 1987 Annual Book of ASTM Standards, Part 11.D18*.
11. AS1289, *Methods of Testing Soils for Engineering Purposes*. Standards Association of Australia.
12. AUSTROADS (1992) *Pavement Design, A Guide to the Structural Design of Road Pavements*, Austroads Publication No AP-17-92, Australian Road Research Board (ARRB).
13. Augusti, G. (1970) Mode Approximations for Rigid-Plastic Structures Supported by an Elastic Medium. *Int. J. Solids Structure*, 6(6), pp. 808-827.
14. Awojobi, A. O., and Gibson, R. E. (1973) Plane Strain and Axially Symmetric Problems of a Linearly Non-homogeneous Elastic Half-space. *Q. J. Mech. Applied Math*, Vol. 26, Part 3.



15. Barden, L. (1963) Stresses and Displacements in a Cross-anisotropic Soil. *Geotechnique*, Vol. 13, No. 3, pp. 198 - 210.
16. Barret, J.R. and Smith, D.M. (1976) *Stress History Effects in Basecourse materials*, Proc. 8<sup>th</sup> Australian Road Research Board (ARRB) Conf. 8(3), Session 7, pp 30-42.
17. Berardi, R., and Lancellotta, R. (1991) Stiffness of Granular Soils from Field Performance. *Geotechnique*, 41, No. 1, pp. 149-157.
18. Borland (1990a) – *Turbo Pascal 6.0 User's Guide*. Borland, USA.
19. Borland (1990b) - *Turbo Pascal 6.0 Programmer's Guide*. Borland, USA.
20. Borland (1990c) - *Turbo Pascal 6.0 Turbo Vision Guide*. Borland, USA.
21. Borland (1990d) – *Turbo Pascal 6.0 Library Reference Guide*. Borland, USA.
22. Borland (1992a) – *Turbo Pascal 7.0 User's Guide*. Borland, USA.
23. Borland (1992b) - *Turbo Pascal 7.0 Language Guide*. Borland, USA.
24. Borland (1992c) - *Turbo Pascal 7.0 Programmer's Guide*. Borland, USA.
25. Borland (1992d) - *Turbo Vision Programming Guide*. Borland, USA.
26. Borowicka, H. (1936) Influence of Rigidity of a Circular Foundation Slab on the Distribution of Pressure over the Contact Surface. *Proceedings of the 1<sup>st</sup> Inter. Conference in Soil Mechanics and Foundation Engineering*, Vol. 2: pp. 144-149.
27. Borowicka, H. (1939) Druckverteilung unter Elastischen Platten, *Ing. Archiv* 10(2), pp. 113-125.
28. Boussinesq, J. (1885) *Application des Potentiels a l'Equilibre et de Mouvement des Solides Elastiques*, Gauthier-Villars, Paris.
29. Bowles, J.E. (1987) Elastic Foundation Settlements on Sand Deposits, *Journal of Geotechnical Engineering*, 113, No 8, August, pp. 846-860.
30. Bowles, J.E. (1997) *Foundation Analysis and Design*, Fifth edition, McGraw-Hill
31. Brown, P.T. (1969) Numerical Analysis of Uniformly Loaded Circular Rafts on Deep Elastic Foundations. *Geotechnique*, 19, pp. 399-404.
32. Brown, P.T., and Gibson, R. E. (1973) Rectangular Loads on Inhomogeneous Elastic Soil. *ASCE*, Vol. 99, No. SM 10, pp. 917- 920.
33. Brown, P.T. (1975) The Significance of Structure-Foundation Interaction. 2<sup>nd</sup> *Australia-New Zealand Conference on Geomechanics*, (Brisbane), pp. 79-82.
34. Bull, J.W. (Edt.) (1994) *Soil-Structure Interaction: Numerical Analysis and Modelling*. E & FN Spon London.
35. Burland, J.B., and Burbidge, M.C. (1985) Settlement of Foundations on Sand and Gravel. *Proceedings Inst. Civil Engineers*, Part 1, 78, pp. 1325-1381.
36. Carrier, W.D., Planetary and Earth Sciences Division, NASA Manned Spacecraft Centre, and Christian, J. T. (1973) Rigid Circular Plate Resting on a Non-Homogeneous Elastic Half-space. *Geotechnique*, 23, No. 1, pp. 67-84.

37. Cerruti, V. (1882) Ricerche intorno all'equilibrio de corpi elastici isotropi. *Reale Accademia dei Lincei*, Vol. 13, Rome, Italy.
38. Chamecki, S. (1956) Structural Rigidity in Calculating Settlements. *J. Soil Mechanics and Foundation Eng. Div. ASCE*, Vol. 82, S.M. 1, 1.
39. Chandrashekhara, K., and Joseph Antony, S. (1993) Theoretical and Experimental Investigation of Framed Structure-Layered Soil Interaction Problems. *Computers and Structures*, Vol. 48, No 2, pp. 263-271.
40. Chandrashekhara, K., and Joseph Antony, S. (1996) Interaction Analysis of Strip Footing on a Non-homogeneous Elastic Medium. *Computers and Structures*, Vol. 60, No 1, pp. 79-86.
41. Cheung, Y.K., and Zienkiewicz, O.C. (1965) Plates and Tanks on Elastic Foundations- an Application of Finite Element Method. *Int. J. Solids Structures*, Vol 1, pp. 451-461.
42. Conte, E., and Dente, G. (1993) Settlement Analysis of Layered Soil Systems by Stiffness Method. *J. Geotechnical Eng.*, ASCE, Vol. 119, No 4, pp. 780-785.
43. Conway, H.D., (1955) Analysis of Some Circular Plates on Elastic Foundations and the Flexural Vibration of Some Circular Plates. *Journal of Applied Mechanics*, 22, pp. 275-276.
44. Davis, E.H., and Poulos, H.G. (1968) The Use of Elastic Theory for Settlement Prediction under Three-dimensional Conditions. *Geotechnique*, 18(1), pp. 67-91.
45. De Beer, E. E. (1967) Bearing capacity and settlement of shallow foundations on sand. *Proc. Symp. on Bearing Capacity and Settlement of Foundations*, Duke University, Durham, North Carolina, pp. 15-33.
46. Delgado, R.M., and Faria, R.M.C.M. (1994) Dynamic Response of Arch Dams with Foundation Interaction. *Soil-Structure Interaction: Numerical Analysis and Modelling*, in: Bull, J.W., (ed.) E & FN Spon London, Chapman & Hall Inc., pp. 167-193.
47. Dempsey, J.P., and Li, H. (1995) A Flexible Rectangular Footing on a Gibson Soil: Required Rigidity for Full Contact. *International Journal of Structures*, Vol. 32, No. 314, pp. 357-373.
48. Desai, C.S., Phan, H.V., and Perumpral, J.V. (1982). Mechanics of three-dimensional soil-structure interaction. *Journal of Structure Division*, ASCE, 104(5), pp. 749-761.
49. Dieterman, H.A., and Metrikine, A.V. (1996) The Equivalent Stiffness of a Half-space Interfacing with a Beam, Critical Velocities of a Moving Load along the Beam. *European J. of Mechanics A/Solids*, 15, pp. 67-90.
50. Dooley, J.C. (1964) Correspondence. *Geotechnique*, 14, No 3, pp. 278-279.

51. Egorov, K.E. (1958) Concerning the Question of the Deformation of Bases of Finite Thickness. *Mekhanika gruntov, Sb. Tr. No 34*, Gosstroizdat, Moscow (in Russian)
52. Fadum, R.E. (1948) Influence Values for Estimating Stresses in Elastic Foundations. *Proceedings of 2<sup>nd</sup> International Conference in Soil Mechanics and Foundation Engineering*, Vol. 3, pp. 77-84.
53. Fedá, J. (1978) *Stresses in Subsoil and Methods of Final Settlement Calculation*. Elsevier Scientific Publishing Co., NY.
54. Florin, V.A. (1961) Fundamentals of soil mechanics. *Gosstroizdat*, Moscow, Vol. 2, p 543.
55. Filonenko-Borodich, M.M. (1940) Some Approximate Theories of the Elastic Foundation. *Uchenyie Zapiski Moskovskogo Gosudarstvennogo Universiteta. Mekhanika*, No. 46, pp. 3-18 (in Russian).
56. Foster, C. R., and Ahlvin, R. G. (1954) Stresses and Deflections Induced by a Uniform Circular Load. *Proceeding of Highway Research Board*, Vol. 33, pp. 467-470.
57. Francis, A.J. (1954) Some Recent Developments in Structural Analysis and Design, *Proceedings Engineering. Conference Melbourne*, Inst. Eng. Australia.
58. Fraser, R.A., and Wardle, L.J. (1976) Numerical Analysis of Rectangular Rafts on Layered Foundations. *Geotechnique*, 26, No. 4, pp. 613-630.
59. Frohlich, O. K. (1934) Druckverteilung im Baugrunde (Distribution of Pressures in soils). *J. Springer-Verlag*, 185, Berlin.
60. Gazetas, G., and Mylonakis, G. (1998) Seismic Soil-structure Interaction: New evidence and emerging issues. in: Yegian, M., and Holtz, R. (Eds.) *Geotechnical Earthquake Engineering and Soil Dynamics III*, University of Washington, Seattle, WA, ASCE, pp. 1119-1174.
61. Gerrard, C.M., (1967) Stresses and Displacements in Layered Cross-anisotropic Elastic System. *Proc. 5<sup>th</sup> Aust.-New Zealand Conf. Soil Mech Foundation Engineering.*, p205
62. Gerrard, C.M., and Harrison, W.J. (1970a) Circular Loads Applied to a Cross-anisotropic Half-space. *C.S.I.R.O. Australia Division of Applied Geomechanics*, Technical Paper No. 8.
63. Gerrard, C.M., and Harrison, W.J. (1970b) Stresses and Displacements in a Loaded Orthohombic Half-space. *C.S.I.R.O. Australia Division of Applied Geomechanics*, Technical Paper No. 9.
64. Gerrard, C.M., Wardle L.J. (1973) Solutions for Point Loads and Generalised Circular Loads Applied to a Cross-anisotropic Half-space, *C.S.I.R.O. Australia Division of Applied Geomechanics*, Technical paper No. 13.

65. Ghali, A., and Neville, A.M. (1989) *Structural Analysis: A Unified Classical and Matrix Approach*. Chapman and Hall, London, Third edition.
66. Gibson, R.E. (1967) Some Results Concerning the Displacements and Stresses in a Non-homogenous Elastic Half-space. *Geotechnique*, Vol. 17, No. 1, pp. 58-67.
67. Gibson, R.E., and Kalsi, G.S. (1974) The Surface Settlement of a Linearly Inhomogeneous Cross-anisotropic Elastic Half-space. *Z. Angew. Math. Phys.*, 25, pp. 843-847.
68. Gibson, R.E., and Sills, G.C. (1975) Settlement of a Strip-load on a Non-homogenous Orthotropic Incompressible Elastic Half-space. *Quart. J. Mech. Appl.*, 28, 2, pp. 233-243.
69. Giroud, J.P. (1968) Settlement of a Linearly Loaded Rectangular Area. *Journal of Soil Mechanics and Foundations Division*, ASCE, Vol 94, No SM44, pp. 813-831.
70. Giroud, J.P. (1970) Stresses under Linearly Loaded Rectangular Area. *Journal of Soil Mechanics Foundations Division*, ASCE Vol 96, SM1, pp. 263-268.
71. Giroud, J.P. (1972) *Tables pour le calcul des fondations*. Tome 1, Tassement, Dunod, 360 p.
72. Giroud, J.P. (1973) *Tables pour le calcul des fondations*. Tome 2, Tassement, Dunod, 505 p.
73. Gorbunov Passadov, M.I. (1949) *Beams and slabs on elastic foundations*. Mashstroizdat, Moscow.
74. Gorbunov Passadov, M.I., Malikova, T.A., and Solomin, V.I. (1984) *Design of structures on elastic foundations*. (in Russian)
75. Goschy, B. (1978) Soil-foundation-structure Interaction, *Journal of Structure*, ASCE, v 104, ST5, May.
76. Griffith, J.H. (1929) Pressures under Substructures. *Engineering and Construction*, Bulletin 117, Iowa State College Eng. Exp. Station, Dynamics of Earth and other Macroscopic Matter.
77. Hain, S., and Lee, I.K. (1980) Prediction of Structure-Foundation Interaction Behaviour. *Proceedings, Vol. 2, 3<sup>rd</sup> Australia-New Zealand Conference on Geomechanics*, The Institution of Engineers, Australia, May, pp. 221-225.
78. Harr, M. E. (1966) *Foundations of Theoretical Soil Mechanics*. McGraw Hill.
79. Hemsley, J.A. (1987) Elastic Solutions for Axisymmetrically loaded circular raft with free or clamped edge founded on Winkler springs or a Half-space. *Proc. Inst. Civil Engineers*, 83, Part 2, Mar. pp. 61-90.
80. Hemsley, J.A. (1988) Comparative Flexure of an Infinite Plate in Winkler springs and a Half-space. *Proc. Inst. Civil Engineers*, 85, Part 2, Dec. pp. 665-685.
81. Hearmon, R. S. F. (1961) *Applied Anisotropic Elasticity*, Oxford University Press.

82. Heaton, B. S. and Bullen, F. (1980) Properties of Air-cooled slag road base for rational pavement design, *Proc. 10<sup>th</sup> ARRB Conf.* 10(3), pp. 198-204.
83. Hetenyi, M. (1946) *Beams on Elastic Foundations*, University of Michigan Press, Ann Arbor, Michigan.
84. Hetenyi, M. (1950) A General Solution for the Bending of Beams on an Elastic Foundation of Arbitrary Continuity. *J. App. Phys.*, Vol. 21, pp. 55-58.
85. Holl, D.L. (1940) Stress Transmission in Earth. *Proceeding of Highway Research Board*, Vol 20, pp. 709-721.
86. Hooper, J.A. (1975) Elastic Settlement of a Circular Raft in Adhesive Contact with a Transversely Isotropic Medium. *Geotechnique*, 25(4), pp. 691-711.
87. Hsieh, Y., and Mau, S.T. (1995) *Elementary Theory of Structures*, Prentice Hall Int. Inc.
88. Hu, Y., Randolph, M.F., and Watson, P.G. (1999) Bearing Response of Skirted Foundation on Non-homogeneous Soil. *J. of Geotechnical and Geoenvironmental Engineering*, Vol. 125, No 11, Nov., pp. 924-935.
89. Hu, Y., Randolph, M.F. (1998) A Practical Numerical Approach for Large Deformation Problems in Soil. *Int. J. for Numerical and Analytical Methods in Geomechanics*. Vol. 22, pp 327-350.
90. Huang, C.C., Menq, F.Y. (1997) Deep-footing and Wide-slab Effects on Reinforced Sandy Ground. *J. of Geotech. And Geoenv. Engineering*, Vol. 123, No 1, Jan. ASCE, pp. 30-36.
91. Huang, C.C., and Tatsuoka, F. (1988) Prediction of Bearing Capacity in Level Sandy Ground Reinforced with Strip Reinforcement. *Proc. Int. Geot. Symp. Theory and Prac. of Earth Reinforcement*, Balkema, Fukuoka, Kyushu, Japan, pp. 619-624.
92. Ishkova, A.G. (1957) On Some Genaralisation of the Solution of the Problems Concerning a Circular Plate and an Infinite Strip on the Elastic Half-space. *Prikl. Mat. Mekh.*, 21, pp. 287-290 (in Russian).
93. I.T.S. Integrated Technical Softwares Pty. Ltd. (1999), *Space Gass 8.01c/32 for Windows*.
94. Izadnegahdar, R. (1998) *Soil And Structure Interaction Package (SASIAP) (Version 1.0) Manual*. Computer Analysis Package, School of Engineering, University of Tasmania.
95. Izadnegahdar, R., and Melerski, E.S. (1997) Analysis of Framed Structures Founded on Elastic Media. Presented article at *the 15<sup>th</sup> Australasian Conference on The Mechanics of Structures and Materials 8th -10th December 1997 in Melbourne University, Proceeding*, pp. 499-504.

96. Kany, M. (1964) *Baugrundverformungen infolge waagerechter Schub-belastung der Baugrundoberfläche*, Die Bautechnik, Heft 10.
97. Kausel, E., and Roesset, J.M. (1981) Stiffness Matrices for Layered Soils. *Bull. of Seism. Soc. of Am.*, 71(6), pp. 1743-1761.
98. Kelvin, Sir William Thomson Baron (1848) Kelvin Solutions, *Cambridge & Dublin Math. J.*
99. Kerr, A.D. (1964) Elastic and Visco-elastic Foundation Models. *J. Applied Mech. Trans.*, ASME, 31, pp. 491-498.
100. Kezdi, A. (1964) *Bodenmechanik*, Band 2, Verl., f., Bauw., Berlin, 459 p.
101. Khing, K.H., Das, B.M., Puri, V.K., Cook, E.E., and Yen S.C. (1992) Bearing Capacity of Two Closely-spaced Strip Foundations on Geogrid-reinforced Sand. *Proc. Earth Reinforcement Prac.*, Balkema, Fukuoka, Kuyushu, Japan, pp. 619-624.
102. Koning, H. (1957) Stress Distribution in a Homogenous, Anisotropic Elastic, Semi-infinite Solid. *Proceeding 4th Conference on Soil Mechanics and Foundation Engineering*. Vol. 1, pp. 335-338.
103. Lambe, T. W. and Whitman, R. V. (1979) *Soil Mechanics, SI Version*, John Wiley & Sons, Inc, New York.
104. Lay, M.G. (1984) Source Book For Australian Roads. *Australian Road Research Board* (ARRB).
105. Lee, I.K. (1969) *Lectures in analysis of settlement of foundations*. University of Sydney.
106. Lee, I.K. (1975) Structure-Foundation-Supporting Soil Interaction Analysis. in: Valliappan, S., Hain, S., and Lee, I.K. (eds.) *Soil Mechanics: Recent Developments*. Proceedings of the General Session of the Symposium held at The University of New South Wales, Australia, July 14-18, pp. 255-294.
107. Lee, I.K., and Brown, E.T. (1972) Structure-Foundation Interaction Analysis. *Journal Structure Division*, ASCE Vol. 98, No, ST 11, November, pp. 2413-2431.
108. Lee, I.K., and Hain, S.J. (1974) Influence of Structural Rigidity on the Performance of Foundations. in: Valliappan, S., Hain, S., and Lee, I.K. (eds.) *Soil Mechanics: Recent Developments*. Proceedings of the General Session of the Symposium held at The University of New South Wales, Australia, July 14-18 p. 294.
109. Lee, I.K., and Harrison, H.B. (1970) A theoretical study of the interaction of structure and foundation. *Journal Structure Division*, ASCE Vol. 96, No, ST2.
110. Lee, I.K., and Valliappan, S. (1974) Analysis of Soil Settlement. In: *Soil Mechanics – New Horizons*, (ed.) Lee I. K. Newnes-Butterworths, London.

111. Lekhnitskii, S.G. (1963) *Theory of Elasticity of an Anisotropic Elastic Body*. Holden-Day Inc. San Francisco.
112. Lempriere, B.M. (1968) Poisson's Ratio in Orthotropic Materials. *J. Am. Inst. Aeronaut Astronaut*, 6 No. 11.
113. Leonards, G. A. & Frost, J. D (1988). Settlement of shallow foundations on granular soils. *J. Geotech. Engineering, Am. Soc. Civ. Engrs* 114, pp. 791-809.
114. Love, A.E.H. (1892) *A Treatise on Mathematical Theory of Elasticity*. Cambridge Eng.
115. Love, A.E.H. (1927) *A Treatise on the Mathematical Theory of Elasticity*. Cambridge Press, 4<sup>th</sup>, 643p.
116. Maier, G., and Novati, G. (1987) Boundary Element Elastic Analysis of Layered Soils by a Successive Stiffness Method. *Int. Journal for Numerical and Analytical Methods in Geomechanics*, 11(5), pp. 435-447.
117. Masih, R. (1985). Computation of Interaction of Structure and Foundation Settlement, *Proc. 2<sup>nd</sup> Int. Conf. On computing in civil engineering*, ASCE, pp. 1048-56.
118. Masih, R. (1993) Structure Stiffness Influence on Soil Consolidation. *Journal of Geotechnical Engineering*, ASCE, Vol. 119, No1, Jan. pp 168-72.
119. Maugeri, M., Castelli, F., Massimino, M.R., and Verona, G. (1998) Observed and Computed Settlements of Two Shallow Foundations in Fand. *J. of Geotechnical and Geoenv. Engineering*, Vol. 124, No. 7, July, pp. 595-605.
120. Meek, J.W., and Wolf, J.P. (1992) Insight on 2D Versus 3D Modelling of Surface Foundations. *Proc. 10<sup>th</sup> World Conf. On Earthquake Engineering*, Madrid, Vol. 3, pp. 1633-1638.
121. Melan, E. (1932) *Der spannungszustand der Durch eine Einzelkraft im Innern Beanspruchten Halbschiebe*, Z. Angew. Math. Mech., 12.
122. Melerski, E S. (1992) Probabilistic Computer Analysis of Circular Rafts by the Method of Realizations. *Computers and structures*, Vol. 43, No 6, pp. 1075-1083.
123. Melerski, E.S. (1993) Comparison of Finite Element and Finite Difference Solutions of Circular Plates and Rafts. *13<sup>th</sup> Australasian Conference on the Mech. of Structures and Materials*, University of Wollongong.
124. Melerski, E.S. (1994) Simplified Numerical Simulation of Interaction Between Circular Rafts and Cross-anisotropic Media. *Civil-Comp.*, Ltd. Edingburg, Scotland, pp. 77-87.
125. Melerski, E.S. (1995a) An Elasto-static Numerical Analysis of Cylindrical Tanks. *14<sup>th</sup> Australasian Conference on the Mechanics of Structures and Materials*.

126. Melerski, E.S. (1995b) Finite Element Analysis of a General Shape Mat Foundation Resting on an Elastic Half-space. *14<sup>th</sup> Australasian Conference on the Mechanics of Structures and Materials*.
127. Menard, L. (1971). Les tassements des foundations et les methode pressiometriques. *Ann. Inst. Technique Batiment Trav. Publ., Paris, No. 288, pp. 105-121*.
128. Metrikine, A.V., and Dieterman, H.A. (1997) The Equivalent Vertical Stiffness of an Elastic Half-space Interacting with a Beam, Including the Shear Stresses at the Beam-half-space Interface. *European J. Mech. A/Solids*, 16, No 3, pp. 515-527.
129. Meyerhof, G.G. (1947) *The Settlement Analysis of Building Frames*. Structural Engineering, Vol. 25, p369.
130. Meyerhof, G.G. (1962) Load-carrying Capacity of Concrete Pavements. *J. Soil Mechanics Foundations Division*, Proceedings ASCE, 88(SM3), pp. 89-116.
131. Meyerhof G. G. (1965). Shallow foundations. *J. Soil Mech. Fndn. Engineering , Am. Soc. Civ. Engrs* 91, No.SM2, pp. 21-31
132. Michell, J.H. (1900) *Stresses and Displacements in an Aeolotropic Solid with an Infinite Plane Boundary*, Proceedings London Mathematic Society, 32, pp. 247-258.
133. Microsoft Corporation (1997) *Microsoft Excel 97 SR-2*
134. Milovic, D. (1992) *Stresses and Displacements for Shallow Foundations*. Elsevier Science Publishers B.V in Geotechnical Engineering.
135. Milovic, D., and Djogo, M. (1991) Settlement of Circular Foundation of any Rigidity. *X European Conference Soil Mech. Found. Eng., Firenze*.
136. Mindlin, R.D. (1936) Force at a Point in the Interior of a Semi-infinite Solid. *J. Applied Physics*, Vol. 7, No. 5, pp. 195-202.
137. Montrasio, L., and Nova, R. (1997) Settlements of Shallow Foundations on Sand: Geometrical Effects. *Geotechnique*, 47, No 1, pp. 49-60.
138. Mulholland, P.J. (1984) *Analysis of Summer Data Collected from Victorian Test Sites*. Australian Road Research Board (ARRB) AIR 392-2.
139. Myslivec, A., Eichler, J., and Jesenak, J. (1970) *Soil Mechanics. (in Czech)* SNTL + Alfa, 388 pp.
140. Newmark, N.M. (1935) Simplified Computation of Vertical Pressures in Elastic Foundations. *University of Illinios Eng. Expt. Stn., Circ. No 24, Vol. 33, No. 4*.
141. Newmark, N.M. (1942) Influence Charts for Computation of Stresses in Elastic Soils. *University of Illinois, Engineering Expt. Stn., Bull. No. 338*.
142. Newmark, N.M. (1947) Influence Charts for Computation of Vertical Displacements in Elastic Soils. *University of Illinois, Engineering Expt. Stn., Bull. No. 367*.



143. O'Brien, S. (1991) - *Turbo Pascal 6 The Complete Reference*. California, McGraw-Hill.
144. Ohde, J. (1939) *Zur theorie der Druckverteilung im Baugrunde*. Der Bauingenieur, Heft 33.
145. Palmer, D.P. (1991) - *Mastering Turbo Pascal 6*. California, Sybex Inc.
146. Papadopoulos, B. (1992) Settlements of Shallow Foundations on Cohesionless Soils. *J. of Geote. Engineering*, Vol. 118, No 3, March, pp. 377-392.
147. Pasternak, P.L. (1954) *On a New Method of Analysis of an Elastic Foundation by Means of two Foundation Constants*. Gosudarstvennoe Izdatelstvo Literaturi po Stroitelstvu i Arkhitekture, Moscow (in Russian).
148. Pickering, D.J. (1970) Anisotropic Elastic Parameters in Soil. *Geotechnique*, 20, pp. 271-276.
149. Pires, J.A., and Higgins, C.J. (1998) Soil-structure Interaction in Cross-anisotropic Site Conditions: in Dakoulas, P., Yegian, M., and Holtz, R. (Eds.) *Geotechnical Earthquake Engineering and Soil Dynamics III*, University of Washington, Seattle, WA, ASCE, pp. 1271-1282.
150. Potter, D. W. (1977) Manual of Operation for Australian Road Research Board (ARRB), wave propagation equipment for the measurement of elastic moduli, *Aust. Rd. Res. Board*, Tech. Manual, ATM No 2.
151. Poulos, H.G. (1975a) Settlement Analysis of Structural Foundation System. 4<sup>th</sup> *Southeast Asian Conference on Soil Engineering*, Kuala Lumpur, 7<sup>th</sup>-10<sup>th</sup> April, Malaysia, pp. 54-62.
152. Poulos, H.G. (1975b) Settlement of Isolated Foundations. in: Valliappan, S., Hain, S. and Lee, I.K (eds.) *Soil mechanics: Recent developments*, Proc. of the General Session of the Symposium UNSW, July, pp. 181-212.
153. Poulos, H.G. (1967) The Use of Sector Method for Calculating Stresses and Displacements in an Elastic Mass. *Proceedings of 5<sup>th</sup> Australian-New Zealand Conference in Soil Mechanics and Foundation Engineering*, pp. 198-204.
154. Poulos, H.G., and Davis, E.H. (1974) *Elastic Solutions for Soil and Rock Mechanics*, John Wiley & Sons Inc. Sydney.
155. Quinlan, P. M. (1949) A Fourier Integral Approach to an Aelotropic Medium. Ph.D. Thesis, California Institute of Technology.
156. Scala, A. J. (1978) A Systems Approach to Flexible pavement design in Australia, *Proc. 9<sup>th</sup> ARRB Conf.*, 9(4), pp. 24-37.
157. Schofield, G. (1986). *Design and Maintenance of Residential Street Pavements*. Master of Engineering Thesis, Royal Melbourne Institute of Technology.
158. Schleicher, F. (1926) *Kreisplatten auf elastischer unterlage*, Julius Springer, Berlin.

159. Schmertmann, J. H. (1970). Static cone to compute settlement over sand. *J. Soil Mech. Fndn Engineering Div., Am. Soc. Civ. Engrs* 96, No SM3, pp. 1011-1043.
160. Scott, R.F. (1963) *Principles of Soil Mechanics*, Addison-Wesley, Reading (Mass.), 550 pp.
161. Seed, H.B., and Idriss, I.M. (1967) Analysis of Liquefaction: Niigata Earthquake. *Proc. ASCE*, Vol. 93, No SM3, pp. 83-108.
162. Seed, H.B., Lysmer, J. and Hwang, R. (1975) Soil-structure Interaction Analyses for Seismic Response. *Journal of the Geotechnical Engineering Division*, Proc. ACSE, Vol. 101, GT5, May, pp. 439-475.
163. Selvadurai, A.P.S (1979) *Elastic Analysis of Soil-Foundation Interaction*. Elsevier, Amsterdam.
164. Shen, W.Y., Chow, Y.K., and Yong, K.Y. (1997) A Variational Approach for Vertical Deformation Analysis of Pile Groups. *Int. J. Num. Analy. Meth. Geomech.*, 21, No 11, pp. 741-752.
165. Shen, W.Y., Chow, Y.K., and Yong, K.Y. (1999) Variational Solution for Vertically Loaded Pile Groups in an Elastic Half-space. *Geotechnique*, 49, No 2, pp. 199-213.
166. Skempton, A.W., and Bjerrum, L. (1957) A Contribution to the Settlement Analysis of Foundations on Clay, *Geotechnique*, vol. 7, p 168.
167. Smith, R.B. and Pratt, D.N. (1983) *A Field Study of in Situ California Bearing Ratio and Dynamic Cone Penetrometer for Road Subgrade Investigations*, Australian Road Research 13 (4).
168. Small, J.C., and Booker, J. R. (1984) Finite Layer Analysis of Layered Elastic Materials Using a Flexibility Approach, Part 1- Strip Loading. *Int. J. for Numerical Methods in Engineering*, 20(6), pp. 1025-1037.
169. Sneddon, I.N. (1951) *Fourier Transformations*, McGraw Hill New York.
170. Sparks, G.H. and Potter, D.W. (1982) *An Investigation into the Relation Between CBR and Modulus for Two Clays*. AIR 295 (1).
171. Steinbrenner, W. (1934) *Tafeln Zur Setzungberchnung*, Strasse. Vol. 1.
172. Stewart, J.P., Fenves, G.L. and Seed, R.B. (1999a) Seismic Soil-Structure Interaction in Buildings, I: Analytical Methods. *Journal of Geot. And Geoenv. Engg.*, Jan. pp 26-37.
173. Stewart, J.P., Fenves, G.L. and Seed, R.B. (1999b) Seismic Soil-Structure Interaction in Buildings, II: Empirical Findings. *Journal of Geot. And Geoenv. Engg.*, Jan. pp 38-48.
174. Suklje, L. (1963) The Equivalent Elastic Constants of Saturated Soils Exhibiting Anisotropy and Creep Effects. *Geotechnique*, Vol. 13, pp. 291-309.

175. Takemura, J., Okamura, M., Suesmasa, N., and Kimura, T. (1992) Bearing Capacity and Deformations of Sand Reinforced with Geogrids, *Proc. Earth Reinforcement Prac.*, Balkema, Fukuoka, Kuyushu, Japan, pp. 695-700.
176. Taylor, D. W. (1948) *Fundamentals of Soil Mechanics*, John Wiley & Sons New York.
177. Terzaghi, K. (1943) *Theoretical Soil Mechanics*. Wiley, New York.
178. Terzaghi, K. (1955) Evaluation of Coefficients of Subgrade Reaction, *Geotechnique*, Vol. 5, pp. 297-326
179. Terzaghi, K. and Peak, R. B. (1967) *Soil mechanics in engineering practice (2nd edn.)* New York: Wiley.
180. Thompson, M.R. and Quentin, L.R. (1976) *Resilient Properties of Subgrade Soils Final Report*. Project IHR-063. Illinois Co-operative Highway and Transportation Research Program, University of Illinois.
181. Timoshenko, S.p. (1934) *Theory of Elasticity*. Mc-Graw Hill, New York.
182. Timoshenko, S.P., and Woinowsky-Krieger, S. (1959) *Theory of Plates and Shells*. McGraw Hill Book Co. Inc., New York.
183. Tranter, C.J. (1966) *Integral Transforms in Mathematical Physics*. Methuens Monographs on Physical Subjects, 3<sup>rd</sup> ed.
184. Urena, R.De., Piquer, J.S., Muzas, F., and Sanz Saz Saracho, J.M. (1966) Stress Distribution in Cross-anisotropic Media. *Proceedings 1<sup>st</sup> Congress Int. Society Rock Mechanics Lisbon*, Vol. 1, pp. 313-317.
185. USBR - United States Department of the Internal Bureau of Reclamation (1956) *Treaties on dams*. Denver, Colorado.
186. Uzan, J., Livneh, M. and Ishai, I. (1980) Thickness Design of flexible pavements with Different Layer Structures, *Aust. Rd. Research*, 10(1), pp. 8-20.
187. Valliappan, S., Hain, S., and Lee, I.K. (eds.) (1975) *Soil Mechanics: Recent Developments*. Proceedings of the General Session of the Symposium held at The University of New South Wales, Australia, July 14-18.
188. Vlasov, V.Z., and Leontiev, U. N. (1960) *Beams, Plates and Shells on Elastic Foundations*. Gosstroizdat Moscow (in Russian).
189. Vlasov, V.Z., and Leontiev, U. N. (1966) *Beams, Plates and Shells on Elastic Foundations*. Israel Program for Scientific Translations, Jerusalem.
190. Vogt, F. (1925) *Ueber die Brerechnung der Fundament Deformation*. Det Norske Videnskaps Akademi, Oslo.
191. Wahls, H.E. (1994) Tolerable Deformations, Vertical and Horizontal Deformations of Foundation and Embankments. *Geothech. Spec. Publ.*, No. 40, Vol. 2, ASCE, pp. 1611-1628.

192. Wardle, L.J. (1977) *Program CIRCLY User's Manual*. CSIRO Division of Applied Geomechanics.
193. Wardle, L.J., and Fraser, R.A. (1974) *Finite Element Analysis of a Plate on a Layered Cross-anisotropic Foundation*. Proc. On Finite Element Methods in Engineering, University of New South Wales, Australia.
194. Watson, G.M. (1944) *The Theory of Bessel Functions*. Cambridge Univ. Press. 2<sup>nd</sup> Ed.
195. Westergaard, H.M. (1938) *A Problem of Elasticity Suggested by a Problem in Soil Mechanics: A Soft Material Reinforced by Numerous Strong Horizontal Sheets*. Mechanics of Solids, S. Timoshenko Sixtieth Anniversary Volume, Macmillan, New York.
196. Winkler, E. (1867) *Die Lehre von Elastizitat und Festigkeit*, Prag, Dominicus, pp. 182.
197. Wolf, K. (1935) Ausbreitung der Kraft in der Halbebene und im Halbraum bei Anisotropem Material. *ZAMM*, 15(5), pp. 249-254.
198. Wolf, J.P., and Meek, J.W. (1994) Insight on 2D versus 3D Modelling of Surface Foundations via Strength of Materials Solutions for Soil Dynamics. *Earthquake Engineering and Structural Dynamics*, Vol. 23, pp. 91-112.
199. Zeleznikov, (1963) (see Milovic 1992).

# APPENDICES

# Appendix A1

## APPENDIX A1

### A1. VERTICAL DISPLACEMENT UNDER A LOAD DISTRIBUTED OVER A CIRCLE AND A RECTANGLE

Two different methods for distributing a concentrated load are presented in this section. Integration of the vertical displacement due to the load over an infinitesimal area is used to obtain the total displacement. The areas considered are a circle and a rectangle.

- A circle (radius  $r_0$ )

Using Equation 2.7, the vertical displacement under the load due to a vertical load can be determined via integration. It is assumed that the load is uniformly distributed over the area (Figure A1.1). The Boussinesq expression is applied to a small area and is integrated. This gives the relationship shown in equation A1.1

$$v_a = \frac{(1-\nu_s^2)}{\pi E_s} \int \frac{dP}{r} = \frac{(1-\nu_s^2)}{\pi E_s} p \int_0^{2\pi} \int_0^{r_0} \frac{r dr d\theta}{r} = \frac{2(1-\nu_s^2)}{E_s} p r_0 \quad (\text{A1.1})$$

when  $P_a = \pi r_0^2 p$  is substituted into Equation A1.1, Equation A1.2 is obtained.

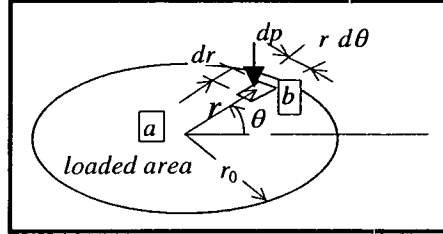


Figure A1.1. Parameters used for a distributed load over a circle.

$$v_a = \frac{2(1-\nu_s^2)}{\pi E_s r_0} P_a = f_{aa}^{ii} P_a \quad (\text{A1.2})$$

- A rectangle ( $a \times b$ )

Cheung and Zienkiewicz (1965) presented a technique involving a uniform distribution of a vertical load over a rectangle (Figure A1.2). The deflection at any point due to a point load is given by the Boussinesq Equation 3.20.

The deflection at the centre of the uniformly loaded rectangular area ( $a \times b$ ) is obtained by integrating the Boussinesq equation over the area

$$u_i = 2 \int_{\zeta=0}^{\zeta=a/2} 2 \int_{\eta=0}^{\eta=b/2} \frac{P_i(1-\nu_s^2)}{ab\pi E_s} \frac{d\zeta d\eta}{\sqrt{\zeta^2 + \eta^2}} = \frac{P_i(1-\nu_s^2)}{a\pi E_s} \chi_{aa}^{ii} \tag{A1.3}$$

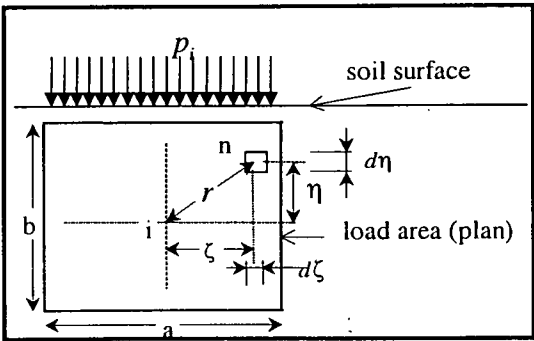


Figure A1.2 Surface vertical displacement due to a uniformly loaded rectangular area on elastic half-space (Cheung and Zienkiewicz 1965).

In the integration, a rectangle  $a \times b$  with a range of aspect ratios ( $b/a$ ) was considered. The resulting values of parameter  $\chi_{aa}^{ii}$  associated with the vertical displacement at the centre of the rectangle for different aspect ratios are given in Table A1.1 (Cheung and Zienkiewicz 1965).

b/a	2/3	1	2	3	4	5
$\chi_{aa}^{ii}$	4.265	3.525	2.406	1.867	1.543	1.322

Table A1.1 Coefficient for deflection at the centre of rectangle  $a \times b$



# Appendix A2

## APPENDIX A2

### A2.1. HOMOGENEOUS ELASTIC HALF-SPACE

#### A2.1.1. Isotropic Elastic Half Space

Mindlin's solutions for the vertical and horizontal displacements for a point within the medium (Figure A2.1) are given below. The displacements due to vertical load  $P$  are as follows:

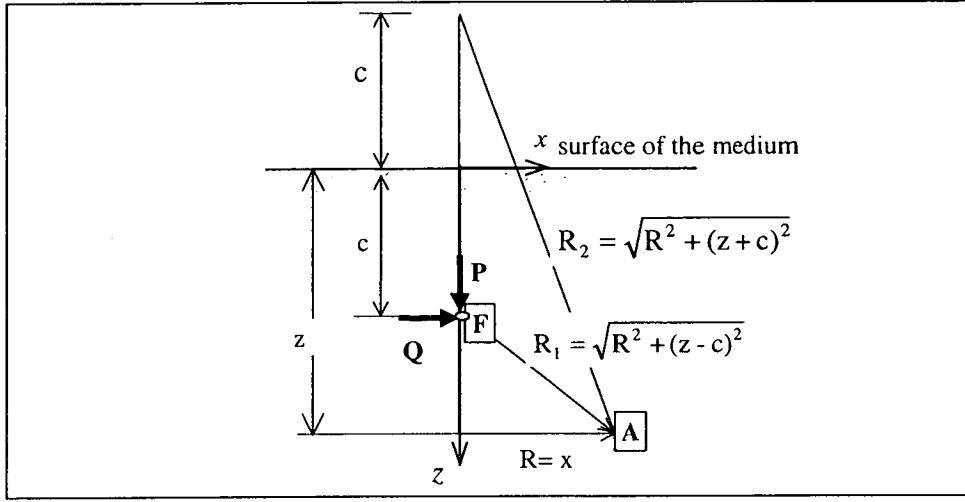


Figure A2.1 Illustration of the Mindlin's case in  $x$ - $z$  plane within the medium.

$$u = \frac{PR}{16\pi G_s(1-\nu_s)} \left[ \frac{z-c}{R_1^3} + \frac{(3-4\nu_s)(z-c)}{R_2^3} - \frac{4(1-\nu_s)(1-2\nu_s)}{R_2(R_2+z+c)} + \frac{6cz(z+c)}{R_2^5} \right] \quad (\text{A2.1})$$

$$w = \frac{P}{16\pi G_s(1-\nu_s)} \left[ \frac{3-4\nu_s}{R_1} + \frac{8(1-\nu_s)^2 - (3-4\nu_s)}{R_2} + \frac{(z-c)^2}{R_1^3} + \frac{(3-4\nu_s)(z+c)^2 - 2cz}{R_2^3} + \frac{6cz(z+c)^2}{R_2^5} \right] \quad (\text{A2.2})$$

where  $G_s$  is the soil shear modulus (in a homogeneous isotropic half-space), and is defined:

$$G_s = \frac{E_s}{2(1+\nu_s)} \quad (\text{A2.3})$$

where  $\nu_s$  is the soil's Poisson's ratio. The displacements due to horizontal load  $Q$  are given:

$$u = \frac{Q}{16\pi G_s(1-\nu_s)} \left[ \frac{(3-4\nu_s)}{R_1} + \frac{1}{R_2} + \frac{x^2}{R_1^3} + \frac{(3-4\nu_s)x^2}{R_2^3} + \frac{2cz}{R_2^3} \left(1 - \frac{3x^2}{R_2^2}\right) + \frac{4(1-\nu_s)(1-2\nu_s)}{R_2+z+c} \left(1 - \frac{x^2}{R_2(R_2+z+c)}\right) \right] \quad (\text{A2.4})$$

$$w = \frac{Qx}{16\pi G_s(1-\nu_s)} \left[ \frac{(z-c)}{R_1^3} + \frac{(3-4\nu_s)(z-c)}{R_2^3} - \frac{6cz(z+c)}{R_2^5} + \frac{4(1-\nu_s)(1-2\nu_s)}{R_2(R_2+z+c)} \right] \quad (\text{A2.5})$$

All the parameters in the above equations are shown in the Figure A2.1.

### A2.1.2. Cross-anisotropic Elastic Half-Space

The following constants are defined in order to simplify the original soil elasticity constants:

$$a = \frac{E_h(1-\nu_{hv}\nu_{vh})}{(1+\nu_h)(1-\nu_h-2\nu_{hv}\nu_{vh})} \quad (\text{A2.6})$$

$$b = \frac{E_h(\nu_h+\nu_{hv}\nu_{vh})}{(1+\nu_h)(1-\nu_h-2\nu_{hv}\nu_{vh})} \quad (\text{A2.7})$$

$$c = \frac{E_h\nu_{vh}}{(1-\nu_h-2\nu_{hv}\nu_{vh})} \quad (\text{A2.8})$$

$$d = \frac{E_h\nu_{vh}(1-\nu_h)}{\nu_{hv}(1-\nu_h-2\nu_{hv}\nu_{vh})} \quad (\text{A2.9})$$

$$f = F_v \text{ soil shear modulus} \quad (\text{A2.10})$$

### A2.1.2.1. Dirac Delta Function:

In the case of application of a point force or moment on a cross-anisotropic medium Dirac delta function is used to define load distribution.

Briefly, the Dirac delta function is a pseudo function that is defined so that it has a defined value everywhere in a domain except at one point. In this approach, the function is expressed as

$$\delta(x) = 0 \text{ for } x \neq 0 \quad (\text{A2.11a})$$

and

$$\int_{-\infty}^{\infty} f(x)\delta(x)dx = f(0) \quad (\text{A2.11b})$$

The function  $\delta(x)$  defined in this way is not like the ordinary functions of mathematical analysis, which are defined to have definite values at each point in certain domains. This form of function is used only when it is obvious that no inconsistency will follow from its use. This option facilitates the integration procedure at the point of load application. The application of the Dirac delta function to the different loads is as follows.

The vertical force load ( $P_z$ ) through the Dirac delta function  $f_a(r)$ , for the vertical stress is distributed such that

$$\sigma_z(r, \theta, 0) = \begin{cases} f_a(r) & r < r_0 \\ 0 & r > r_0 \end{cases} \quad (\text{A2.12a})$$

Then

$$f_a(r) = \frac{P_z \delta(r)}{2\pi r} \quad (\text{A2.12b})$$

Hence

$$P_z = 2\pi \int_0^{r_0} r \cdot f_a(r) dr \quad (\text{A2.12c})$$

The appropriate Hankel transform simplifies to

$${}_aH_0(k) = \frac{P_z}{2\pi} \quad (\text{A2.12d})$$

Horizontal force load ( $Q_x$ ) through the Dirac delta function  $f_b(r)$ , for the uni-directional stress, is distributed such that

$$\tau_{xz}(r, \theta, 0) = \begin{cases} f_b(r)r < r_0 \\ 0 & r > r_0 \end{cases} \quad (\text{A2.13a})$$

Then

$$f_b(r) = \frac{Q_x \delta(r)}{2\pi r} \quad (\text{A2.13b})$$

Hence

$$Q_x = 2\pi \int_0^{r_0} r \cdot f_b(r) dr \quad (\text{A2.13c})$$

The appropriate Hankel transform simplifies

$${}_b H_0(k) = \frac{Q_x}{2\pi} \quad (\text{A2.13d})$$

Finally, the horizontal moment load ( $M_y$ ), through the Dirac delta function  $f_c(r)$ , for the linearly vertical stress, is distributed such that

$$\sigma_z(r, \theta, 0) / \cos \theta = \begin{cases} f_c(r)r < r_0 \\ 0 & r > r_0 \end{cases} \quad (\text{A2.14a})$$

Then

$$f_c(r) = \frac{2 M_y \delta(r)}{2\pi r^2} \quad (\text{A2.14b})$$

And

$$\tau_{rz}(r, \theta, 0) = \tau_{\alpha z}(r, \theta, 0) = 0 \quad (\text{A2.14c})$$

Hence

$$M_y = \pi \int_0^{r_0} r^2 \cdot f_c(r) dr \quad (\text{A2.14d})$$

And the appropriate Hankel transform simplifies

$${}_c H_1(k) = \frac{M_y}{2\pi} k \quad (\text{A2.14e})$$

### A2.1.2.2. Constants and Coefficients Associated with Numerical Integration Method

Generally, the external load is described by a stress distribution function over a circular area with radius  $R$ . Such a function is expressed either in the form of  $(1 - R^2)^q$  or  $R(1 - R^2)^t$  with parameters  $q$  and  $t$  are obtained accordingly (Sneddon 1951, Tranter

1966, Gerrard and Harrison 1970a, and Gerrard and Wardle 1973). In general, the solutions are products of coefficients ( $g_1, \dots, g_4, h_1, \dots, h_4, h_{10}, i_1, \dots, i_4, j_1, \dots, j_4, j_{11}, s_1, \dots, s_4, t_1, \dots, t_4, t_{10}$ ) and integrals ( $I, A$ , and  $M$  types). The coefficients are functions only of the elastic properties while the integrals are in general functions of depth, radial offset and elastic parameters. The following integrals are used to develop solutions for displacements:

$$I_{\eta\tau\mu}(\Psi) = \int_0^\infty J_{0.5\eta}(K) J_{0.5\tau}(KR) K^{0.5(\mu-2)} \exp(-\Psi K) dK \quad (\text{A2.15a})$$

$${}_c I_{\eta\tau\mu} = \int_0^\infty J_{0.5\eta}(K) J_{0.5\tau}(KR) K^{0.5(\mu-2)} \cos \omega KZ \exp(-\alpha KZ) dK \quad (\text{A2.15b})$$

$${}_s I_{\eta\tau\mu} = \int_0^\infty J_{0.5\eta}(K) J_{0.5\tau}(KR) K^{0.5(\mu-2)} \sin \omega KZ \exp(-\alpha KZ) dK \quad (\text{A2.15c})$$

$$A_{\eta\tau\mu}(\Psi) = \frac{1}{2} [I_{\eta(\eta+2)\mu}(\Psi) + I_{\eta(\tau-2)\mu}(\Psi)] \quad (\text{A2.15d})$$

$$M_{\eta\tau\mu}(\Psi) = \frac{1}{2} [I_{\eta(\eta+2)\mu}(\Psi) - I_{\eta(\tau-2)\mu}(\Psi)] \quad (\text{A2.15e})$$

$${}_s M_{\eta\tau\mu}(\Psi) = \frac{1}{2} [{}_s I_{\eta(\eta+2)\mu} + {}_s I_{\eta(\tau-2)\mu}] \quad (\text{A2.15f})$$

Integrals of products of Hankel transform kernels, Bessel functions, trigonometric and exponential functions are employed to find the solutions for the displacement functions. The load coefficients are presented in their generalised form as follows:

$$L_{\zeta\tau\lambda}(\psi) = \int_0^\infty H_\zeta(k) J_{0.5\tau}(kr) k^{0.5\lambda} \exp(-\psi k) dk \quad (\text{A2.16a})$$

$${}_c L_{\zeta\tau\lambda} = \int_0^\infty H_\zeta(k) J_{0.5\tau}(kr) k^{0.5\lambda} \exp(-\alpha kz) \cos \omega kz dk \quad (\text{A2.16b})$$

$${}_s L_{\zeta\tau\lambda} = \int_0^\infty H_\zeta(k) J_{0.5\tau}(kr) k^{0.5\lambda} \exp(-\alpha kz) \sin \omega kz dk \quad (\text{A2.16c})$$

$$P_{\zeta\tau\lambda}(\psi) = \frac{1}{2} L_{\zeta(\tau+2)\lambda}(\psi) + \frac{1}{2} L_{\zeta(\tau-2)\lambda}(\psi) \quad (\text{A2.16d})$$

$$S_{\zeta\tau\lambda}(\psi) = \frac{1}{2} L_{\zeta(\tau+2)\lambda}(\psi) - \frac{1}{2} L_{\zeta(\tau-2)\lambda}(\psi) \quad (\text{A2.16e})$$

$${}_c S_{\zeta\tau\lambda} = \frac{1}{2} {}_c L_{\zeta(\tau+2)\lambda} - \frac{1}{2} {}_c L_{\zeta(\tau-2)\lambda} \quad (\text{A2.16f})$$

$${}_s S_{\zeta\tau\lambda} = \frac{1}{2} {}_s L_{\zeta(\tau+2)\lambda} - \frac{1}{2} {}_s L_{\zeta(\tau-2)\lambda} \quad (\text{A2.16g})$$

The following relations were defined to come up with the expression of displacements in a cross-anisotropic medium in section 2.4 of Chapter2.

$$\phi = \alpha - \beta \quad (\text{A2.17a}); \quad \rho = \alpha + \beta \quad (\text{A2.17b})$$

$$\omega^2 = -\beta^2 \quad (\text{A2.18}); \quad \gamma^2 = \frac{(a-b)}{f} \quad (\text{A2.19})$$

$$\psi = \alpha z, \phi z, \rho z, \gamma z \quad (\text{A2.20})$$

$$g_1 = \frac{(2c+f)\rho\phi}{f(\rho-\phi)(c+d\phi^2)} \quad (\text{A2.21}); \quad g_2 = \frac{(2c+f)\rho\phi}{f(\rho-\phi)(c+d\rho^2)} \quad (\text{A2.22})$$

$$g_3 = \frac{(2d\phi^2-f)\rho}{f(\rho-\phi)(c+d\phi^2)} \quad (\text{A2.23}); \quad g_4 = \frac{(2d\rho^2-f)\phi}{f(\rho-\phi)(c+d\rho^2)} \quad (\text{A2.24})$$

$$h_1 = \frac{(2c+f)\phi}{f(\rho-\phi)(c+d\phi^2)} \quad (\text{A2.25}); \quad h_2 = \frac{(2c+f)\rho}{f(\rho-\phi)(c+d\rho^2)} \quad (\text{A2.26})$$

$$h_3 = \frac{(2d\phi^2-f)}{f(\rho-\phi)(c+d\phi^2)} \quad (\text{A2.27}); \quad h_4 = \frac{(2d\rho^2-f)}{f(\rho-\phi)(c+d\rho^2)} \quad (\text{A2.28})$$

$$h_{10} = t_{10} = j_{11} = \frac{2}{f\gamma} \quad (\text{A2.29})$$

$$i_1 = \frac{2\alpha\sqrt{ad}}{ad-c^2} \quad (\text{A2.30}); \quad i_2 = \frac{\sqrt{a}}{f\omega\sqrt{d}} \quad (\text{A2.31})$$

$$i_3 = \frac{1}{\sqrt{ad}+c} \quad (\text{A2.32}); \quad i_4 = -\frac{\alpha}{\omega(\sqrt{ad}-c)} \quad (\text{A2.33})$$

$$j_1 = \frac{1}{\sqrt{ad} + c} \quad (\text{A2.34}); \quad j_2 = \frac{\alpha(\sqrt{ad} - c)}{\omega} \quad (\text{A2.35})$$

$$j_3 = -\frac{2\alpha d}{ad - c^2} \quad (\text{A2.36}); \quad j_4 = \frac{1}{f\omega} \quad (\text{A2.37})$$

$$t_1 = \frac{(2c + f)(d\alpha^2 - c)}{f(c + d\alpha^2)^2} \quad (\text{A2.38}); \quad t_2 = \frac{(2c + f)\alpha}{f(c + d\alpha^2)} \quad (\text{A2.39})$$

$$t_3 = \frac{(2c + f)2d\alpha}{f(c + d\alpha^2)^2} \quad (\text{A2.40}); \quad t_4 = \frac{(2d\alpha^2 - f)}{f(c + d\alpha^2)} \quad (\text{A2.41})$$

$$s_1 = \frac{(2c + f)2d\alpha^3}{f(c + d\alpha^2)^2} \quad (\text{A2.42}); \quad s_2 = \frac{(2c + f)\alpha^2}{f(c + d\alpha^2)} \quad (\text{A2.43})$$

$$s_3 = \frac{(2c + 3f - 2d\alpha^2)d\alpha^2 + cf}{f(c + d\alpha^2)^2} \quad (\text{A2.44}); \quad s_4 = \frac{(2d\alpha^2 - f)\alpha}{f(c + d\alpha^2)} \quad (\text{A2.45})$$

Equations A2.16a to A2.16g are applied to obtain the integral constants, which were used to express the displacements, occurred in the cross-anisotropic medium. The expressions in the form of  ${}_2F_1(a_1, a_2, a_3, a_4)$  are hyper-geometric functions that are described in Equations A2.76 to A2.80.

$$A_{220}(0) = \begin{cases} \frac{1}{2} \times {}_2F_1\left(\frac{1}{2}, \frac{-1}{2}; 2; \frac{1}{r^2}\right) \\ \frac{1}{2r} \times {}_2F_1\left(\frac{1}{2}, \frac{-1}{2}; 2; \frac{1}{r^2}\right) \end{cases} \quad (\text{A2.46})$$

$$I_{200}(0) = {}_c I_{200} = \begin{cases} {}_2F_1\left(\frac{1}{2}, \frac{-1}{2}; 1; r^2\right) \\ \frac{1}{2r} \times {}_2F_1\left(\frac{1}{2}, \frac{1}{2}; 2; \frac{1}{r^2}\right) \end{cases} \quad (\text{A2.47}); \quad {}_s I_{200} = {}_s I_{220} = 0 \quad (\text{A2.48})$$



$${}_s I_{220} = {}_s M_{220} = \begin{cases} 0 \\ 0 \end{cases} \quad (A2.49); \quad I_{420}(0) = {}_c I_{420} = \begin{cases} \frac{1}{2} r \times {}_2F_1\left(\frac{3}{2}, \frac{-1}{2}; 2; r^2\right) \\ \frac{1}{8r^2} \times {}_2F_1\left(\frac{3}{2}, \frac{1}{2}; 3; \frac{1}{r^2}\right) \end{cases} \quad (A2.50)$$

$$I_{422}(0) = {}_s I_{420} = {}_s M_{420} = \begin{cases} 0 \\ 0 \end{cases} \quad (A2.51)$$

$$L_{000}(\psi) = \frac{P_z}{2\pi} \frac{1}{\sqrt{(\psi^2 + r^2)}} \quad (A2.52)$$

$${}_c L_{000} = \frac{P_z}{2\pi} \frac{\cos 0.5 v_p}{\sqrt{R_p}} \quad (A2.53); \quad {}_s L_{000} = \frac{P_z}{2\pi} \frac{\sin 0.5 v_p}{\sqrt{R_p}} \quad (A2.54)$$

$$L_{020}(\psi) = \frac{P_z}{2\pi} \left( \frac{1}{r} - \frac{\psi}{r\sqrt{\psi^2 + r^2}} \right) \quad (A2.55)$$

$${}_c L_{020} = \frac{P_z}{2\pi} \frac{\sqrt{R_p} - \alpha \cos 0.5 v_p - \omega \sin 0.5 v_p}{r\sqrt{R_p}} \quad (A2.56)$$

$${}_s L_{020} = \frac{P_z}{2\pi} \frac{\omega z \cos 0.5 v_p - \alpha z \sin 0.5 v_p}{r\sqrt{R_p}} \quad (A2.57)$$

$$L_{002}(\psi) = \frac{P_z}{2\pi} \frac{\psi}{\sqrt{(\psi^2 + r^2)^3}} \quad (A2.58); \quad L_{022}(\psi) = \frac{P_z}{2\pi} \frac{r}{\sqrt{(\psi^2 + r^2)^3}} \quad (A2.59)$$

$$L_{120}(\psi) = \frac{M_y}{2\pi} \frac{r}{\sqrt{(\psi^2 + r^2)^3}} \quad (A2.60); \quad {}_c L_{120} = \frac{M_y}{2\pi} \frac{r \cos \frac{3}{2} v_p}{\sqrt{R_p^3}} \quad (A2.61)$$

$${}_s L_{120} = \frac{M_y}{2\pi} \frac{r \sin \frac{3}{2} v_p}{\sqrt{R_p^3}} \quad (A2.62); \quad L_{122}(\psi) = \frac{M_y}{2\pi} \frac{3\psi r}{\sqrt{(\psi^2 + r^2)^5}} \quad (A2.63)$$

where

$$R_p = \sqrt{[(\alpha^2 - \omega^2)z^2 + r^2]^2 + 4\alpha^2\omega^2z^4} \quad (A2.64)$$

and

$$\nu_p = \tan^{-1} \left[ \frac{2\alpha \omega z^2}{(\alpha^2 - \omega^2) z^2 + r^2} \right] \quad (\text{A2.65})$$

where  $P_{020}(0)$ ,  $S_{020}(0)$ ,  ${}_cS_{020}$ ,  ${}_sS_{020}$ ,  $S_{022}$ ,  $S_{120}(0)$ ,  ${}_cS_{120}$ ,  ${}_sS_{120}$ , and  $S_{122}(0)$  are the symbols which were used in defining load functions in sections 2.4.1 and 2.4.2 of Chapter 2.

The following relations have been used in section 2.4.2.3 of Chapter 2 to express the displacements occurred due to the applied tangential and vertical loads.

$$M_{220}(0) = {}_cM_{220} = \begin{cases} -\frac{1}{2} \times {}_2F_1\left(\frac{3}{2}, \frac{-1}{2}; 2; r^2\right) \\ -\frac{1}{8r^3} \times {}_2F_1\left(\frac{3}{2}, \frac{1}{2}; 3; \frac{1}{r^2}\right) \end{cases} \quad (\text{A2.66})$$

$${}_cM_{420} = \begin{cases} -\frac{1}{4} \\ \frac{1}{8r^2} \end{cases} \quad (\text{A2.67})$$

$$M_{422}(0) = \begin{cases} \frac{1}{2} \times {}_2F_1\left(\frac{3}{2}, \frac{1}{2}; 2; \frac{1}{r^2}\right) - {}_2F_1\left(\frac{3}{2}, \frac{-1}{2}; 2; r^2\right) \\ \frac{1}{2r^3} \left[ {}_2F_1\left(\frac{3}{2}, \frac{1}{2}; 2; \frac{1}{r^2}\right) - \frac{1}{2} \times {}_2F_1\left(\frac{3}{2}, \frac{1}{2}; 3; \frac{1}{r^2}\right) \right] \end{cases} \quad (\text{A2.68})$$

In a cross-anisotropic medium, when elasticity moduli are such that  $\alpha^2 = 1$  and  $\beta^2 = 0$ , the soil is then simplified to one with isotropic properties, where coefficients  $s_1, s_2, \dots, s_8$  and  $f$  are simplified to the following terms:

$$s_1 = \frac{2(1-\nu_s^2)}{E_s} \quad (\text{A2.69}); \quad s_2 = s_4 = \frac{1+\nu_s}{E_s} \quad (\text{A2.70})$$

$$s_3 = s_9 = \frac{2(1+\nu_s)(1-2\nu_s)}{E_s} \quad (\text{A2.71}); \quad s_5 = s_6 = 1 \quad (\text{A2.72})$$

$$s_7 = 2\nu_s \quad (A2.73); \quad s_8 = 0 \quad (A2.74)$$

$$f = a - b = \frac{E_s}{1 + \nu_s} \quad (A2.75)$$

In integral constants,  ${}_2F_1(a_1, a_2, a_3, a_4)$  expressions are hyper-geometric functions. The numerical values of these functions are obtained either from the terms of their series or the hyper-geometric tables. The formulation to evaluate these expressions is as follows:

$$F(a, b; c; z) = {}_2F_1(a, b; c; z) \quad (A2.76)$$

$$= F(b, a; c; z) = \sum_{n=0}^{\infty} \frac{(a)_n (b)_n}{(c)_n} \frac{z^n}{n!} \quad (A2.77)$$

$$= \frac{\Gamma(c)}{\Gamma(a) \Gamma(b)} \sum_{n=0}^{\infty} \frac{\Gamma(a+n) \Gamma(b+n)}{\Gamma(c+n)} \frac{z^n}{n!} \quad (A2.78)$$

where  $\Gamma(a)$  is a gamma function in general is defined as  $\Gamma(a) = \int_0^{\infty} t^{a-1} e^{-t} dt$ ,  $(a > 0)$ .

Also,

$$\frac{d}{dz} {}_2F_1(a, b; c; z) = \frac{a b}{c} {}_2F_1(a+1, b+1; c+1; z) \quad (A2.79)$$

For the case where  $z = r^2$ , the Equation (A2.79) simplifies

$$\frac{d}{dz} {}_2F_1(a, b; c; r^2) = \frac{2 r a b}{c} {}_2F_1(a+1, b+1; c+1; r^2) \quad (A2.80)$$

## A2.2. METHODS OF OBTAINING SOIL PARAMETERS

Methods to obtain elastic parameters in soil which are used in road and structure designs are based on a number of assessments. California Bearing Ratio (*CBR*) is one of these tests that may be used to determine the modulus of sub-grade reaction, as is applied in the Winkler model. When the sub-grade is in a critical moisture condition a degree of caution in carrying out the tests is required, or alternatively, seasonal adjustments may be made.

### A2.2.1. Determination of Elastic Parameters of an Isotropic Soil

A uniaxial soil test is carried and vertical compression and lateral expansion is obtained such that

$$\varepsilon_z = \frac{\sigma_z}{E} \quad (\text{A2.81})$$

$$\varepsilon_x = \varepsilon_y = -\nu \varepsilon_z \quad (\text{A2.82})$$

where  $\varepsilon_x$ ,  $\varepsilon_y$ ,  $\varepsilon_z$  are strains in the  $x$ ,  $y$ ,  $z$  directions respectively,  $E$  and  $\nu$  are Young's modulus and Poisson's ratio.

If shear stresses  $\tau_{zx}$  are applied to an elastic cube, there will be a shear distortion such that

$$\gamma_{zx} = \frac{\tau_{zx}}{G} \quad (\text{A2.83})$$

where  $G$  is shear modulus. Equations A2.82 and A2.83 define the three basic constants of the theory of elasticity:  $E$ ,  $G$ ,  $\nu$ . In fact, only two of these constants are independent since

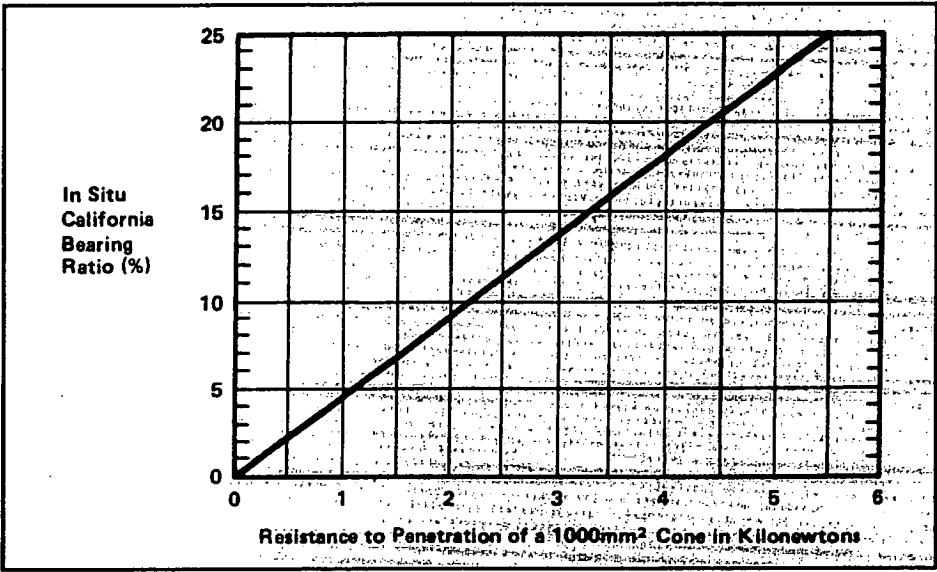
$$G = \frac{E}{2(1 + \nu)} \quad (\text{A2.84})$$

### A2.2.2. In situ Determination of CBR and Elastic Parameters

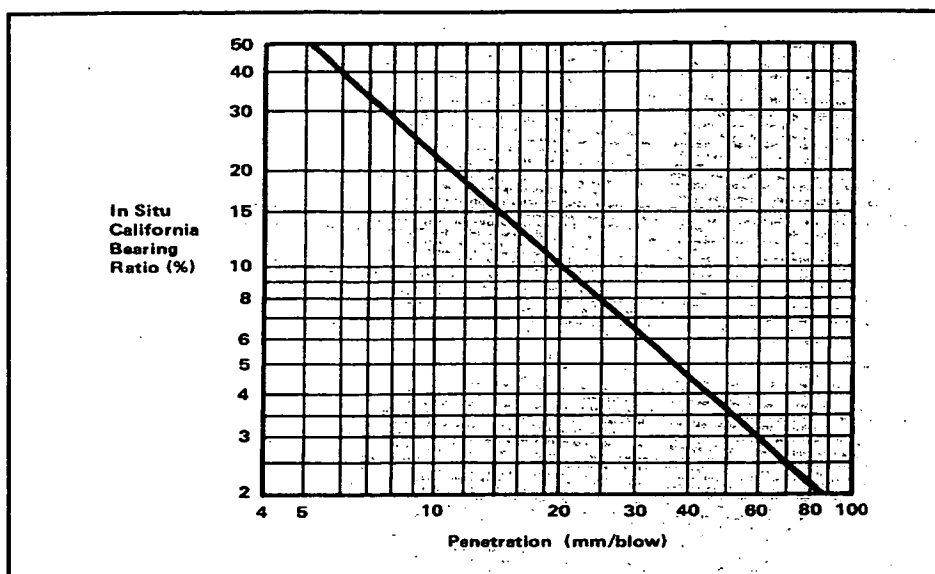
A number of field tests can be used to estimate sub-grade *CBR*, e.g. in situ *CBR* test and Cone Penetrometer. The results of such tests should be statically analysed and a design *CBR* chosen at a percentile level appropriate to the particular case. The ten percentile level value (Mean minus 1.3 times Standard Deviation) is commonly used for design of highway pavements (Austroads, Pavement Design 1992).

The in situ *CBR* test procedure is outlined in AS1289. This test is time-consuming and expensive. The number of tests required to establish the variability of the *CBR* for each type of soil may be so large as to make the use of an in situ *CBR* test impracticable.

Cone Penetrometer tests (AS1289) may be used for fine-grained subgrades. *CBR* results can be determined from Graph A2.1 for a static cone penetrometer and from Graph A2.2 for a dynamic cone penetrometer. The relationship that can be obtained from Graphs A2.1 and A2.2 is a general relation that suits most soil types. Further information regarding specific soil types can be obtained from Smith and Pratt (1983), Mulholland (1984) and Schofield (1986). When using the cone penetrometers extensively for subgrade investigation, a limited number of in situ *CBR* measurements should be carried out on the particular material that are being tested to confirm that the adopted relation is valid.



Graph A2.1 Correlation of static cone penetrometer and CBR (Austroads, Pavement Design 1992)



Graph A2.2 Correlation of dynamic cone penetrometer and CBR (Austroads, Pavement Design 1992)

### A2.2.3. Laboratory Determination of CBR and Soil Elastic Parameters

Laboratory procedure may be used to determine design *CBR* or soil elastic modulus where sufficient samples of the subgrade material can be obtained for detailed laboratory investigations and where a reasonable estimate can be made of likely subgrade density and moisture conditions in service. This method is particularly useful when a close similarity in density, moisture content and material is not available between the proposed pavement and any existing half-space.

Laboratory tests may be undertaken on specimens compacted at the design moisture content (DMC) and density which correspond to those likely to occur in service or at a particular compaction standard and moisture as a characterising test. Alternatively, undisturbed samples can be obtained from the field coring. Further suggestions have been provided by Pavement Design (Austroads 1992) where it is not possible to prepare laboratory specimens at the selected density.

For a cross-anisotropic medium, Wardle (1977) considered that the elastic constants required are three moduli (vertical, horizontal and shear) and two Poisson's ratios. By

taking the ratio of vertical to horizontal modulus to be 2 and if both Poisson's ratios are equal, the following relationship between three of these quantities can be obtained

$$\text{Shear Modulus} = \frac{\text{Vertical Modulus}}{1 + \text{Poisson's Ratio}} \quad (\text{A2.85})$$

Hence, the values of the 5 parameters can be determined from those for vertical modulus and Poisson's ratio.

The vertical modulus of a subgrade can be determined either from the laboratory testing of conditioned specimens (Thompson and Quentin 1976) or from the following empirical relationship (This relationship was mentioned earlier in Chapter 2)(Austroads, Pavement Design 1992)

$$\text{Vertical Modulus of Elasticity (MPa)} = 10 \text{ CBR} \quad (\text{A2.86})$$

This equation is at best an approximation. The modulus has been found to vary in the range 5-20 CBR (Sparks and Potter 1982).

Due to the complexity of soil, even an estimation of the approximate values for soil properties may be difficult. To overcome this issue, a more consistent nature of the unloading curve is utilised, and a resilient (Young's) modulus is defined in terms of the recoverable strain upon unloading (Lay 1984). This modulus depends on the size of the applied stress and the number of prior load applications. The modulus also depends upon material type, moisture content, density soil suction and grading. Barret and Smith (1976) found that the modulus of elasticity for a crushed dolerite with clay binder, from repeated load triaxial tests (Figure A2.1) could be adequately predicted by the following equation

$$E = k (\sigma_1 + 2\sigma_3)^m \quad \text{MPa} \quad (\text{A2.87})$$

where  $k$  and  $m$  are parameters shown in Figure A2.2 and seen to depend on the number of load repetitions,  $\sigma_1$  is the applied compressive stress,  $\sigma_3$  is the orthogonal

restraining stress and  $(\sigma_1 + 2\sigma_3)$  is thus proportional to the mean normal stress defined

$$s = \frac{(\sigma_1 + 2\sigma_3)}{3} \tag{A2.88}$$

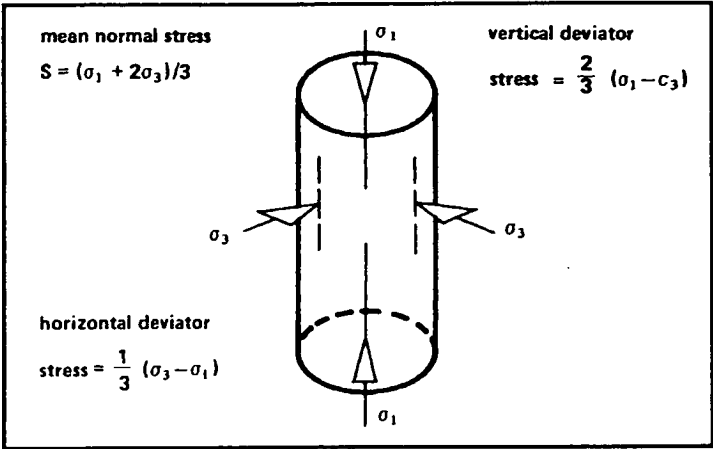
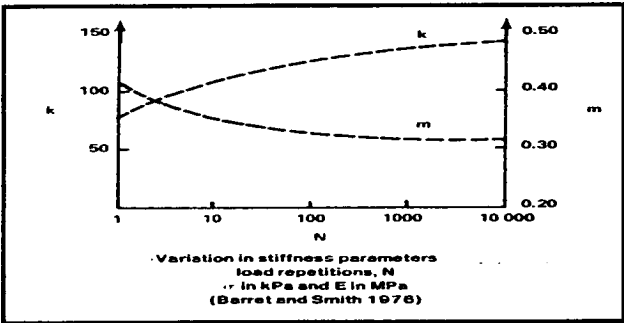


Figure A2.2 Stress definitions in triaxial test

Barret and Smith (1976) showed (Graph A2.3) that the values of  $k$  and  $m$  depend on the number of load repetitions.



Graph A2.3 Variation in stiffness parameters used in equation (2.79) with load repetitions (Barret and Smith (1976)

A maximum value of 150 MPa is normally used for subgrade materials (Austroads, Pavement Design 1992). Variation of  $\nu$  is between 0.1 to 0.50, with the lower limit corresponding to rocks or sands with a high amount of volume changes whereas the upper limit is associated with a saturated soil with almost zero volumetric change under external load materials. Pavement Design (Austroads, 1992) suggests values of Poisson’s ratio for subgrades to be taken 0.45 for cohesive material and 0.35 for non-cohesive material.



### A2.2.4. Adoption of Presumptive CBR Values

When no other relevant information is available, some values for the CBR are assumed. Typical presumptive values of CBR are given in Table A2.1. However, such values should be determined on the basis of local experience.

Description of Subgrade Material	USC Classification	Typical CBR Values % Well drained	Poorly drained
Highly Plastic Clay	CH	5	2~3
Silt	ML		
Silty Clay	CL	6~7	4~5
Sandy Clay	SC		
Sand	SW, SP	15~20	N/A

Table A2.1 Presumptive values of CBR for different types of soil (Austroads, Pavement Design 1992)

Some other researchers have suggested similar equations, e.g. Heaton and Bullen (1980) for blast furnace slag. A hyperbolic relation between stress deviation and permanent strain was suggested by Akili (1980) and stated in the following form

$$\frac{E}{E_{initial}} = \left( 1 - \frac{\text{deviator}}{\text{ultimate stress}} \right)^2 \quad (A2.89)$$

According to suggestions made by Uzan et al. (1980) the modulus of elasticity  $E$  in  $MPa$  is somewhere between 10 and 16 times the CBR in percent. This study leads subgrade  $E$ 's of between 10 and 300  $MPa$ .

Another approach to determining in situ  $E$  values is Deflection bowl analysis. The bowl is determined during Benkelman beam testing, which is normally conducted as part of a pavement evaluation study. Scala (1978) proposed the following equation for the estimation of  $E$  for a two-layer pavement

$$E_1 = 185 \left( \frac{dm}{d_{250}} - 1 \right)^{-1.75} \quad MPa \quad (A2.90)$$

where  $d_m$  is the bowl and usually measured under dual-tyre loading and  $d_{250}$  on the line of travel. He suggested that  $E_1$  for granular bases is reasonably constant at 350 MPa . However, values as low as 100 MPa can be encountered.

The approximate stiffness moduli for a given existing medium can be non-destructively determined using the wave propagation or dynamic modulus method (Potter 1977). In this method, the pavement surface is subjected to vertical vibrations of a known frequency. The velocity of the waves radiating horizontally from the vibrator is determined for a number of different frequencies. By comparing these measurements with theoretical predictions, the elastic moduli for the various pavement layers can be deduced. Use of the method requires knowledge of the number and thickness of any medium layers present. The values obtained by this approach have about 30% accuracy.

# Appendix A3

## APPENDIX A3

### A3. DEVELOPMENT OF STRUCTURE STIFFNESS MATRICES & DIRECT STIFFNESS MATRIX METHOD

The direct stiffness matrix method due to its importance is presented here. Further reference can be made to Ghali and Neville (1989) and Hsieh and Mau (1995). This method is used to analyse a superstructure that lies in  $x - y$  plane, where the forces, the displacements and the translations occur in the same plane, whereas the nodal rotations and the moments occur about the  $z$ -axis, which is perpendicular to the plane  $x - y$ . These axes follow the right hand rule. The applied sign convention is presented in a free-body diagram of the structure element in Figure A3.1.

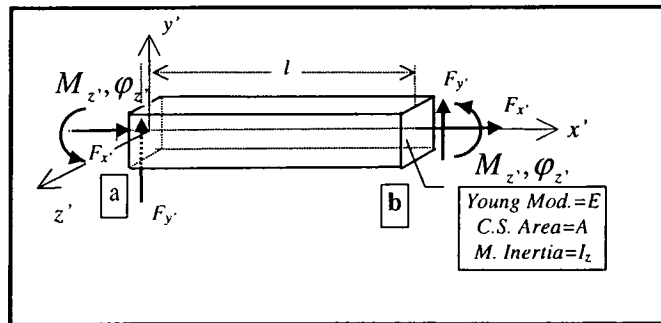


Figure A3.1 Element free-body diagram in local coordinates.

The element stiffness coefficient is represented by  $k_{ab}$ , where a force at point  $a$  corresponds to a displacement at point  $b$ . In general, with any two points  $a$  and  $b$ , there are four possible stiffness coefficients  $k_{aa}$ ,  $k_{bb}$ ,  $k_{ab}$  and  $k_{ba}$ . The first two are known as direct stiffness coefficients and the remaining two are called cross stiffness coefficients.

Steps of the analysis are as follows:

1. Stiffness coefficients for each element (Figure A3.1) are developed and a double subscript  $k_{ab}$  is assigned. The first subscript  $a$ , addresses the location

of the force for which the equation is written, and the second subscript  $b$  refers to the associated degree of freedom.

2. Generally, for each element, the stiffness matrix is generated in local coordinates where both ends are fixed. The matrix stiffness relation is as follows:

$$\begin{Bmatrix} \bar{F}_{x'a} \\ \bar{F}_{y'a} \\ \bar{M}_{z'a} \\ \bar{F}_{x'b} \\ \bar{F}_{y'b} \\ \bar{M}_{z'b} \end{Bmatrix}_e = \begin{bmatrix} \frac{EA}{l} & 0 & 0 & -\frac{EA}{l} & 0 & 0 \\ 0 & \frac{12EA}{l^3} & \frac{6EA}{l^2} & 0 & -\frac{12EA}{l^3} & \frac{6EA}{l^2} \\ 0 & \frac{6EA}{l^2} & \frac{4EA}{l} & 0 & -\frac{6EA}{l^2} & \frac{2EA}{l} \\ -\frac{EA}{l} & 0 & 0 & \frac{EA}{l} & 0 & 0 \\ 0 & -\frac{12EA}{l^3} & -\frac{6EA}{l^2} & 0 & \frac{12EA}{l^3} & -\frac{6EA}{l^2} \\ 0 & \frac{6EA}{l^2} & \frac{2EA}{l} & 0 & -\frac{6EA}{l^2} & \frac{4EA}{l} \end{bmatrix}_e \begin{Bmatrix} u'_a \\ v'_a \\ \theta'_a \\ u'_b \\ v'_b \\ \theta'_b \end{Bmatrix}_e \quad (\text{A3.1a})$$

The element stiffness relation (Equation A3.1a) can be written as

$$\{\bar{F}'\}_e = [k']_e \{\Delta'\}_e \quad (\text{A3.1b})$$

where  $\{\bar{F}'\}_e$  and  $\{\Delta'\}_e$  are the column vector of element nodal forces and the column vector of element nodal displacements, respectively.

Matrix  $[k']_e$  is the element stiffness matrix in the local coordinates, which is symmetric with respect to its main diagonal. Symbols  $l$ ,  $A$ ,  $I$  and  $E$  designated as length, cross-sectional area, moment of inertia about the  $z$ -axis and Young's modulus, respectively. The prime (') indicates the local coordinates for the element, and the bar over the element end forces indicates that these end forces result from the nodal displacements only.

3. Combining the element stiffness matrices according to their nodes and degrees of freedom establishes the superstructure stiffness matrix  $[K]_s$ . Such a combination for a beam structure with three nodes and two elements, where node 2 is common to both elements (1 and 2) is shown in Figure A3.2.

$$[K] = \begin{bmatrix} \{\Delta\}_{(1)} & \{\Delta\}_{(2)} & \{\Delta\}_{(3)} \\ \begin{bmatrix} [k_{11}]_{(1)} & [k_{12}]_{(1)} \\ [k_{21}]_{(1)} & [k_{22}]_{(1)} + [k_{22}]_{(2)} \end{bmatrix} & \begin{bmatrix} 0 \\ [k_{23}]_{(2)} \end{bmatrix} \\ 0 & \begin{bmatrix} [k_{32}]_{(2)} \\ [k_{33}]_{(2)} \end{bmatrix} \end{bmatrix}$$

Figure A3.2 Stiffness matrix for a structure with two members and three nodes.

In Figure A3.2,  $\{\Delta\}_{(i)}$  ( $i = 1, 2, 3$  in this example) represent three columns related to the displacements  $u_i, v_i$  and  $\theta_i$  of node  $i$  and  $[k_{ab}]_{(e)}$  is a  $3 \times 3$  sub-matrix of stiffness matrix for element  $(e)$  associated with the forces at node  $a$  due to displacements of node  $b$ .

In the above figure, the boundaries of the stiffness matrices of the first and the second element are shown by a broken line (---) and a dotted line (.....) respectively.

Each element based on its alignment (angle  $\varphi$ ), with respect to the global coordinates, has its own transformation matrix (Figure A3.3). For a structure composed of elements with different local coordinates, if both the element nodal forces and displacements are expressed in global coordinates, the conditions of both equilibrium of forces and compatibility of displacements at the structure nodes can be applied.

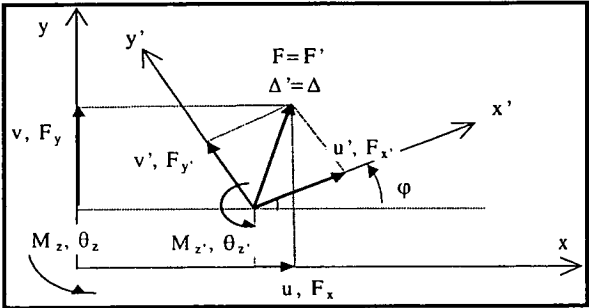


Figure A3.3 Orientation of displacements and forces in both coordinate systems

The following transformation matrix is used for a typical element transformation from global to local coordinates as shown in Figure A3.3.

$$[T]_e = \begin{bmatrix} \cos \varphi & \sin \varphi & 0 & 0 & 0 & 0 \\ -\sin \varphi & \cos \varphi & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \varphi & \sin \varphi & 0 \\ 0 & 0 & 0 & -\sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}_e \quad (\text{A3.2})$$

The transformation of nodal displacements is carried out using the following relationship.

$$\begin{Bmatrix} u'_a \\ v'_a \\ \theta_a \\ u'_b \\ v'_b \\ \theta_b \end{Bmatrix}_e = [T]_e \begin{Bmatrix} u_a \\ v_a \\ \theta_a \\ u_b \\ v_b \\ \theta_b \end{Bmatrix}_e \quad (\text{A3.3a})$$

This equation can be written as

$$\{\Delta'\}_e = [T]_e \{\Delta\}_e \quad (\text{A3.3b})$$

where  $\{\Delta\}_e$  and  $[T]_e$  are the vector of element nodal displacements in the global coordinates and the transformation matrix respectively (see Figure A3.3).

The transformation matrix  $[T]_e$  is orthogonal, therefore, its inverse matrix equals its transpose. Hence,

$$[T]_e^{-1} = [T]_e^T \quad (\text{A3.4})$$

This property facilitates the transformations between the coordinates, which apply to both vectors of forces and displacements. The nodal displacements in the global coordinates are obtained by

$$\{\Delta\}_e = [T]_e^T \{\Delta'\}_e \quad (\text{A3.5})$$

Similarly, the nodal forces can be transformed between the two coordinates as follows:

$$\{\bar{F}'\}_e = [T]_e \{\bar{F}\}_e \quad (\text{A3.6})$$

$$\{\bar{F}\}_e = [T]_e^T \{\bar{F}'\}_e \quad (\text{A3.7})$$

The element stiffness matrix relation in the global coordinates can be obtained as

$$\{\bar{F}\}_e = [k]_e \{\Delta\}_e \quad (\text{A3.8})$$

where

$$[k]_e = [T]_e^T [k']_e [T]_e \quad (\text{A3.9})$$

Matrix  $[k]_e$  is the global element stiffness matrix.

4. When calculating the fixed-end forces, the loads applied between the element nodes that cause the fixed end forces are taken into account. For a member  $a-b$ , the fixed-end forces are

$$\{F^o\} = \begin{Bmatrix} F_{x'a}^o \\ F_{y'a}^o \\ M_{z'a}^o \\ F_{x'b}^o \\ F_{y'b}^o \\ M_{z'b}^o \end{Bmatrix} \quad (\text{A3.10})$$

These fixed end forces are illustrated in Figure A3.4.

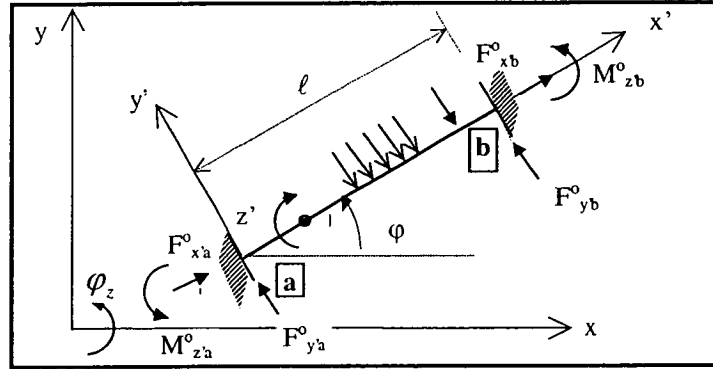


Figure A3.4 Orientation of fixed-end forces in the local coordinates.

The total element end forces can be expressed by combining both the end forces that result from the end displacements with those associated with the loads applied to the member. Thus

$$\{F\}_e = \{\bar{F}\}_e + \{F^o\}_e = [k']_e \{\Delta\}_e + \{F^o\}_e \quad (\text{A3.11})$$



where  $\{F\}_e$  is the vector of total element end forces in local coordinates and  $\{F^0\}_e$  is the vector of fixed end forces due to the loads applied to the element, all in local coordinates. Similarly, in global coordinates

$$\{F\}_e = \{\bar{F}\}_e + \{F^0\}_e = [k]_e \{\Delta\}_e + \{F^0\}_e \quad (\text{A3.12})$$

where

$$\{F^0\}_e = [T]_e^T \{F^0\}_e \quad (\text{A3.13})$$

where  $\{F^0\}_e$  is the vector of global fixed end forces due to loads applied to the element.

5. To account for the support conditions, the soil stiffness matrix is combined with the global structure stiffness matrix according to their degrees of freedom, and the system stiffness matrix is obtained

$$[K]_{sys} = [K]_{st} + [K]_s \quad (\text{A3.14})$$

Then, the system equilibrium matrix equation in global coordinates is

$$\{P\} = [K]_{sys} \{\Delta\} + \{P^o\} \quad (\text{A3.15})$$

where  $\{P\}$  is the vector of the external forces applied to the structure nodes,  $\{P^o\}$  is the vector of equivalent nodal forces for the structure that are obtained by combining the corresponding global fixed-end forces of the individual members  $\{F^0\}$  common at each node, and  $\{\Delta\}$  is the vector of the global system nodal displacements.

Equation (A3.15) can be written in the following form:

$$[K]_{sys} \{\Delta\} = \{P\} - \{P^o\} = \{P\}_{eff} \quad (\text{A3.16})$$

6. The vector of nodal displacements is obtained

$$\{\Delta\} = [K]_{sys}^{-1} \{P\}_{eff} \quad (\text{A3.17})$$

7. Finally, the displacements from Equation A3.17 are substituted back into the individual element force-displacement equations. The result for the  $i^{th}$  element internal forces with corresponding displacements  $\{\Delta\}_{(i)}$  is obtained from the following equation

$$\{F\}_{(i)} = [k]_{(i)} \{\Delta\}_{(i)} + \{F\}_{(i)}^0 \quad (A3.18)$$

Both the internal forces and the nodal displacements are obtained in the global coordinates, and can be transformed to the local coordinates using the transformation matrix. Thus

$$\{F\}_e = [T]_e \{F\}_e \quad (A3.19)$$

# Appendix A4

# **APPENDIX A4**

## **USER MANUAL FOR**

### **SASIAP ANALYSIS PACKAGE**

#### **A4.1. Introduction**

The description given in this manual is a user guide for software package (*Soil And Structure Interaction Analysis Package SASIAP*) discussed in Chapter 4 of the thesis with the aim of only implementation of the basic approach presented in that chapter. Considering this aim and also resources restrictions, developer has set out some initial conditions that user requires attention. The following information introduces user with capability of this package and provides some instruction.

*SASIAP* is a *DOS* application programme. This software like any software has some restrictions. In summary, structural data is validated after being entered and before analysis proceeds. If there is some inconsistency such as conflicting data or structural instability, the programme warns through appropriate dialog boxes accordingly, and after giving advice about the possible problem it quits execution where the situation needs to be amended. A number of flow charts are provided in Chapter 4 to view the locations of these checkpoints.

In summary, operation of this package is based on execution of two modules. The structural data is entered through first module *DATAENTR*. Then, second module *ANALYSIS* analyses the structure according to the options made via interface during execution.

The conditions are as follows:

1. *SASIAP* package is a tool for planar frame structure elastostatic analysis using Direct Stiffness Matrix method, incorporating various member connections, restraints, cross-section modulus and loads applicable to the structure.
2. Before analysis, structural external stability should be checked.

3. Implying half-space, the structural supporting nodes are considered geometrically in the same level.
4. Global coordinates should cover the entire structural nodes in the quarter of plane frame where  $x \geq 0$  and  $y \geq 0$ .
5. Assigning start and end nodes in each member should keep the angle between the member's alignment and the global coordinates in  $-0.5\pi \leq \varphi \leq +0.5\pi$  zone.
6. Structural node numbering should be in an increasing order for a narrower bandwidth and achieving a faster operation.
7. To create blank input structural data files, first *Genforms.bat* should be executed.
8. *DATAENTR* is then run to enable enter, save and retrieve the structural data.
9. Finally, programme *ANALYSIS* analyses the structure contributing soil interaction according to options selected by user, then the output files are saved in different forms.

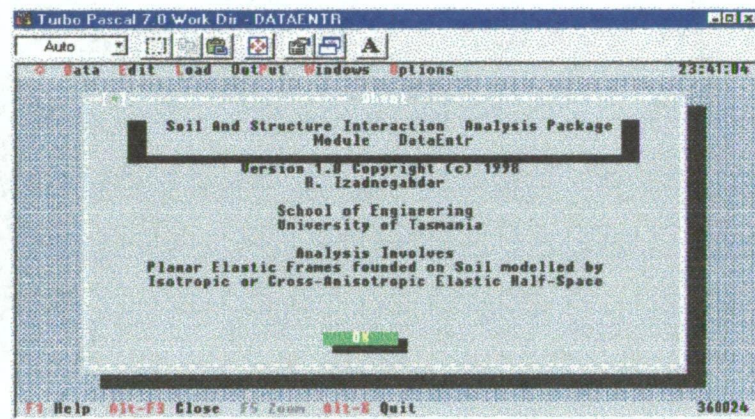


Figure A4.1 illustrates the introduction of DataEntr program

There are several pull-down menus at the top of the menu screen that can be accessed by the mouse, or using “Alt” key combined with the letter key high-lighted in red for that menu (Figures A4.1, ..., A4.30). The arrow keys are used to move up, down or across the pull-down menus. Feature of few menus is described in the following sections.

## A4.2. Data Menu

Data pull-down menu (Figure A4.2) includes the following operations:



- “New” creates a blank text window with a default file named *NoName.DAT* linked to structural and load data, which would be accessed via interface and is associated with individual project (Figures A4.2 and A4.3).

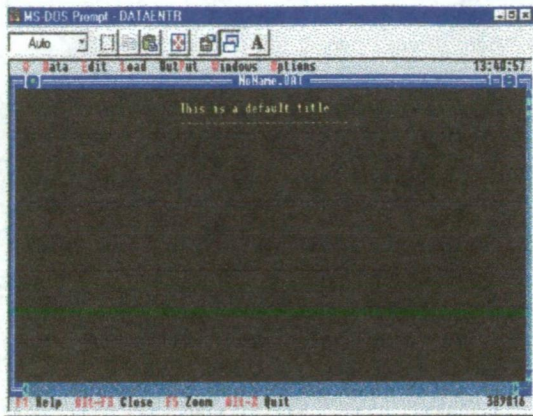


Figure A4.2. A blank text associated with a default title.

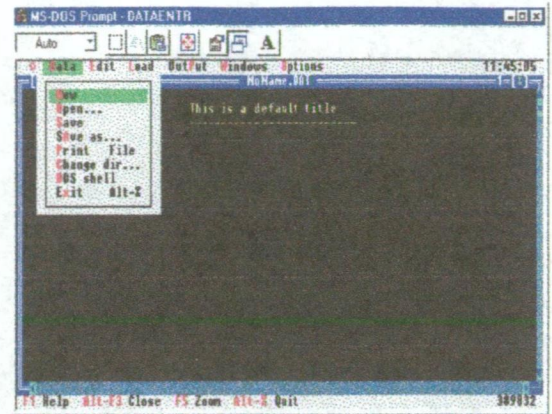


Figure A4.3. Data menu bar and its pull-down items.

- “Open” prepares a window for user with a list of existing files to select for a specific task, and open the required file (Figure A4.4).

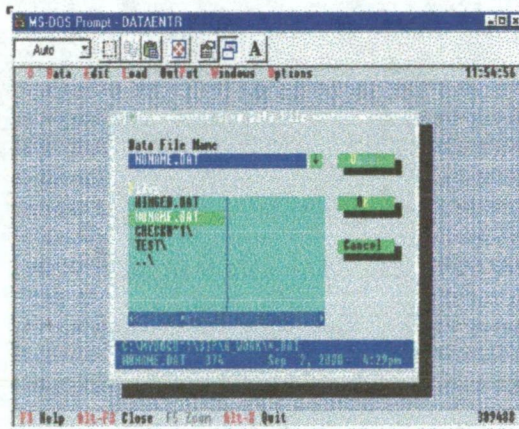


Figure A4.4 Access and open an existing structure data file

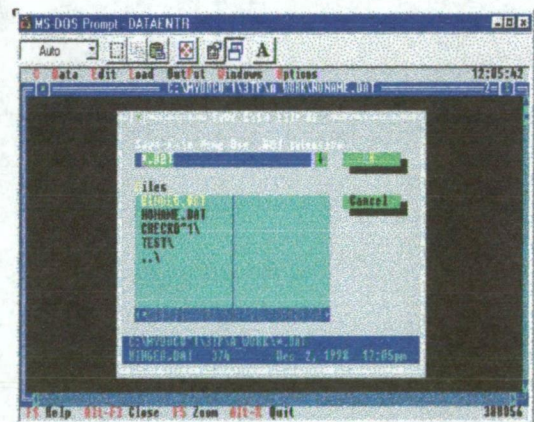


Figure A4.5 Illustration of window *Save As* to save a file under a name other than the original.

- “Save” option writes the content of an open data file (an active window) to disk under the name which a file was opened.
- “Save as ...” saves the active data file in association with a task under a different name with a default extension *DAT* (Figure A4.5).
- “Print File” sends the entire contents of an active window to the printer.

- “Change directory...” opens a dialog box that contains a directory tree with the DOS structure. This option allows the user to change the currently working directory. To select a new directory, a double-click on the destination address is required (Figure A4.6).

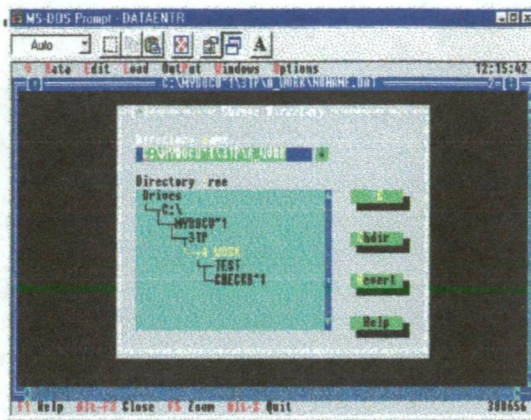


Figure A4.6 illustrates the root and current directories for any new location of a directory

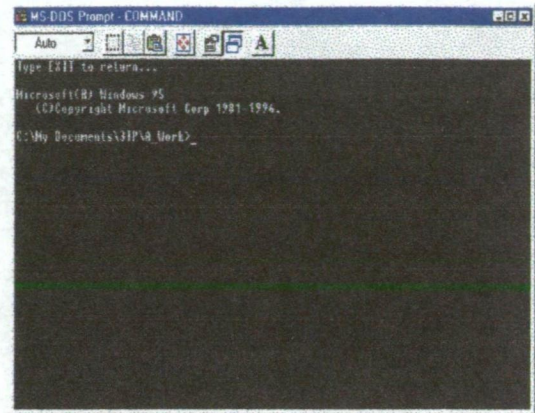


Figure A.7 DOS shell access within the program

- “DOS shell” opens a temporary DOS prompt line within integrated development environment (IDE) where Turbo Pascal environment is operational. By typing “Exit” in the DOS prompt at any time, the user can return back to the DATAENTR’s menu system (Figure A4.7).
- “Exit” quits the user from the DATAENTR programme.



### A4.3. Edit Menu

Upon selecting this pull-down menu, a series of structural data becomes available for edition. This includes “General Data”, “Node Geometry”, “Material Prop”, “CrossSect Spec”, “Member Data” and “Restrains” (see Figure A4.8).

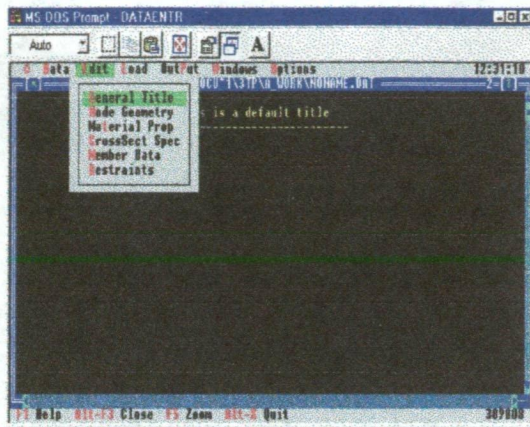


Figure A4.8 illustrates the items of Edit data pull-down menu

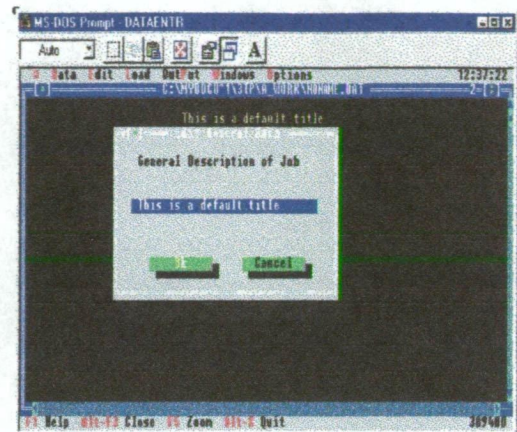


Figure A4.8 Window associated with the title of a structure

By selecting any of the items from this pull-down menu, a directory tree appears that allows the user to open an existing file of structural or load data. Alternatively, by opting for the default blank file associated with a record containing a single member, user may key in the structural data. On quitting, a new name with a default extension (which is strongly recommended) or the existing default name and extension can be assigned to save the data entered. The assignment of individual items of this menu is described below:

- “General Data” a title can be assigned to a set of data that is either a default title or a new one. This title brings all the associated structural and load data files under one cover (incorporates all the corresponding structural data files) (Figure A4.9).
- “Node Geometry” provides access to view or amend node geometry data of a planar structure to be saved or was saved earlier. Node number, global coordinates X and Y, and some space for some description for each node is saved (Figures A4.10 and A4.11).



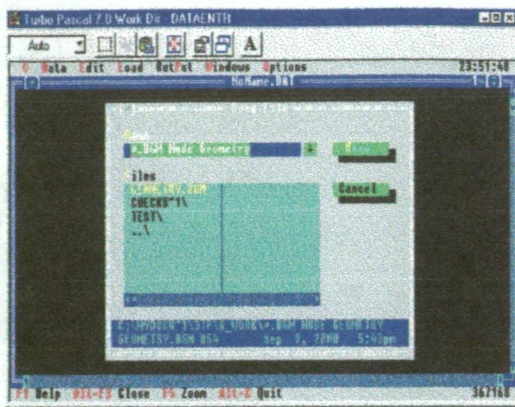


Figure A4.10 Selection of a data file associated with Geometry of a Structure nodes

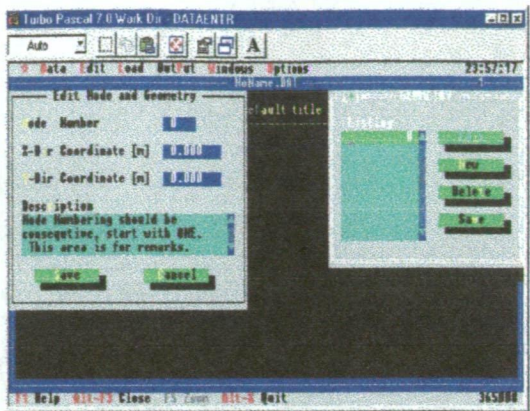


Figure A4.11 Geometry details of a structure node entered or edited

- “Material Properties” provides access to view or amend materials properties data associated with every structural member. This includes material type number, modulus of elasticity, thermal coefficient and some space for relevant description (Figures A4.12 and A4.13).

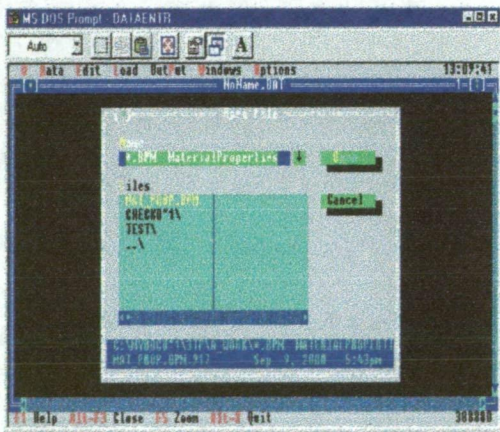


Figure A4.12 Window to access a file of material properties

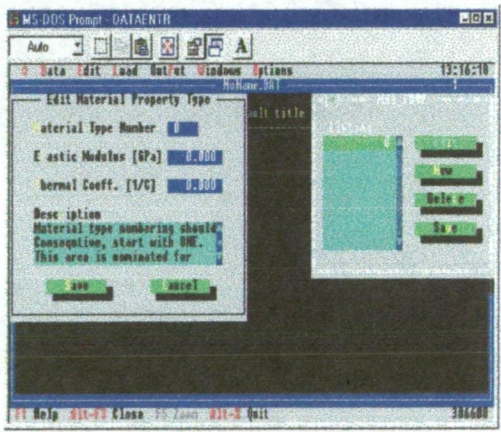


Figure A4.13 Illustration of material properties data window

- “Cross-section Specifications” provides access to view or amend member cross-section data associated with each of the structure member that includes cross section type number, cross section area, moment of inertia about the axis perpendicular to the frame plane, member width and some space for relevant description (Figures A4.14 and A4.15).



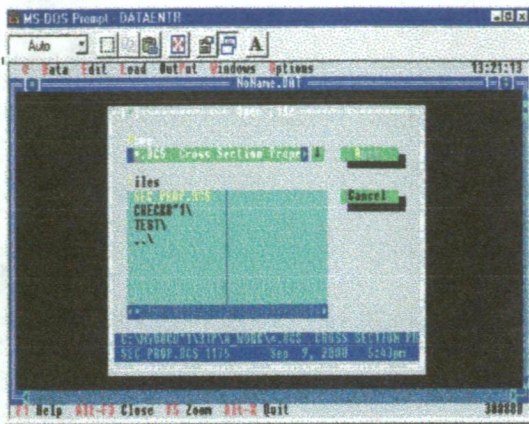


Figure A4.14 allocating a cross section data file

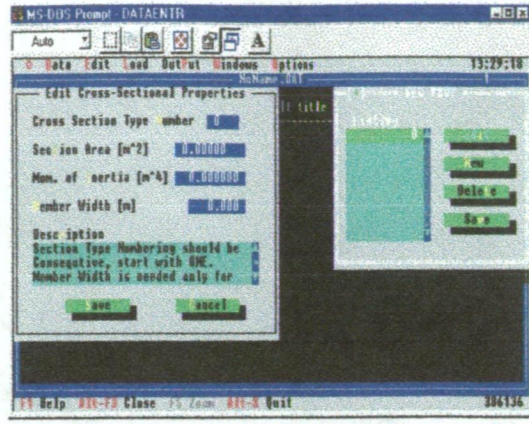


Figure A4.15 Cross-section properties associated with structure members

- “Member Data” enables user retrieve the information associated with each member of a structure. That is member number, start and end node numbers including type of inter-connections such as fixed or hinged, cross-section type, material type and some space for relevant description (Figures A4.16 and A4.17).

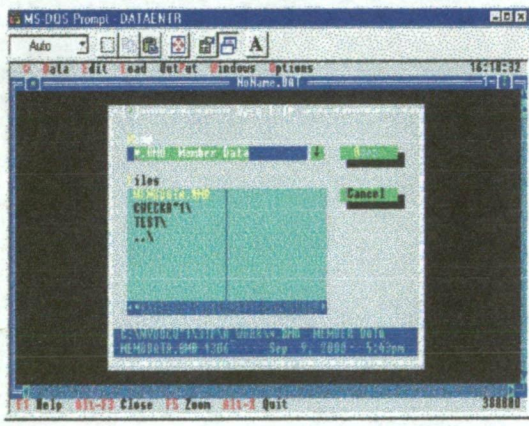


Figure A4.16 Allocating a member data file

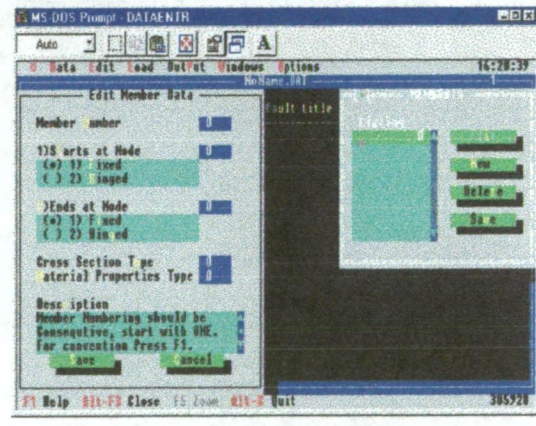


Figure A4.17 Data associated with members of a structure

- “Restraints” provides access to the information related to restraints applicable to structure nodes. This includes number of the node associated with support; type of support applicable in  $X$ ,  $Y$  and  $Z$  directions such as fully restrained, free to take displacement and elastic restraint with a required field for the spring coefficient in  $kPa$  and some space for relevant description (Figures A4.18 and A4.19).



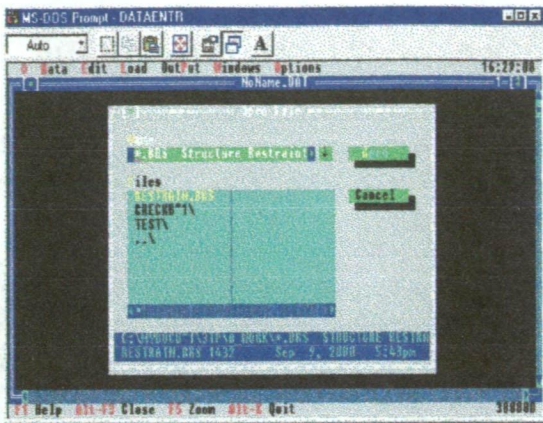


Figure A4.18 Allocating a file for Restrain data.

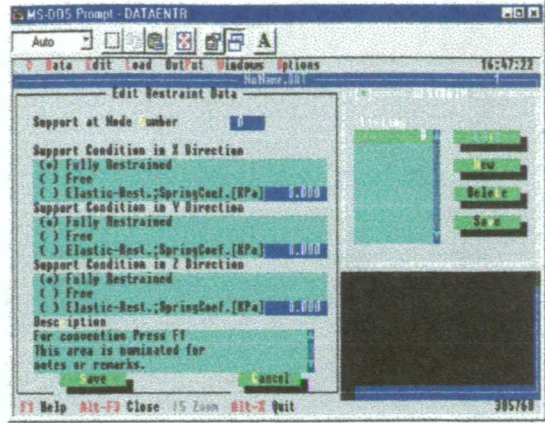


Figure A4.19 Information associated with node restraints

### A4.4. Load Menu

By this pull-down menu, a series of load becomes available that includes “*Prescribed Displacement Load PDL*”, “*Nodal Point Load NPL*”, “*Member Point Load MPL*”, “*Member Distributed Load*” with two options “*Uniform Load MUL*” and “*Trapezoidal Load MTL*” (Figure A4.20).

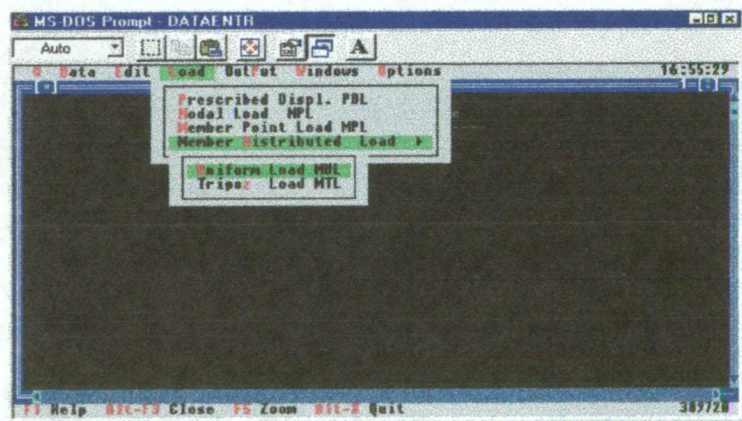


Figure A4.20 Load pull-down menu

By selecting any of the items from this pull-down menu, a directory tree allows the user to open an existing data load file associated with every member of structure for view or edit. On quitting, a new name with a default extension (which is strongly recommended), or the existing default name and extension can be assigned to save the data. The assignment of individual items of this menu is described below:



- “*Prescribed Displacement Load PDL*” user retrieves information regarding implemented prescribed displacements. This information is number of the node where the displacement is applied, amount of displacement along *X*, *Y* directions in meter and rotation about *Z* axis in radian, and also some space for individual technical description; Figures A4.21 and A4.22).

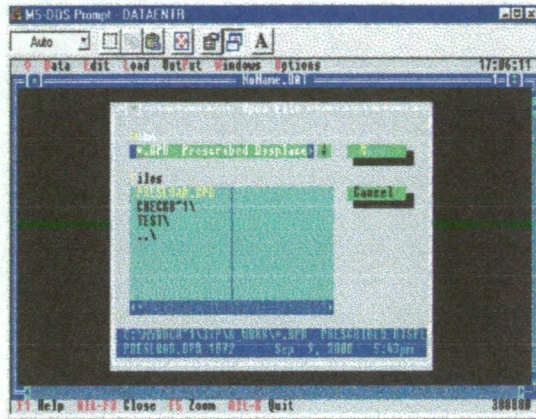


Figure A4.21 Allocating Prescribed Load file.

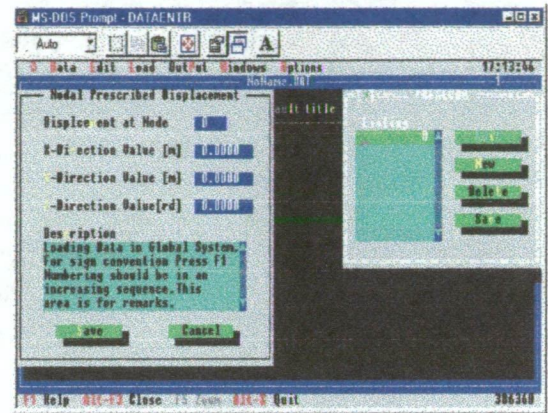


Figure A4.22 Details of prescribed displacement loads applicable to structure nodes

- “*Nodal Point Load NPL*” provides access to information regarding nodal loads. This includes number of node where a concentrated load is applied, amount of force in *X* and *Y* directions in *kN* or moment about *Z* axis in *kN/m* and some space for technical description (Figures A4.23 and A4.24).

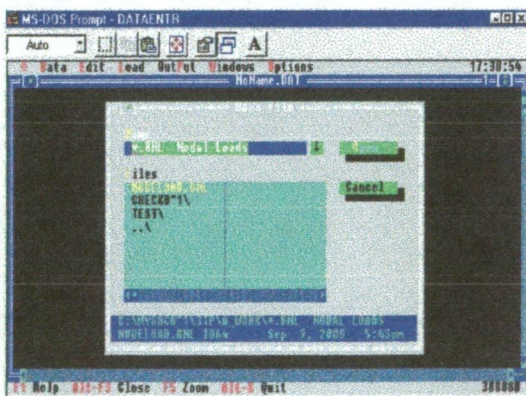


Figure A4.23 Allocating Node load file

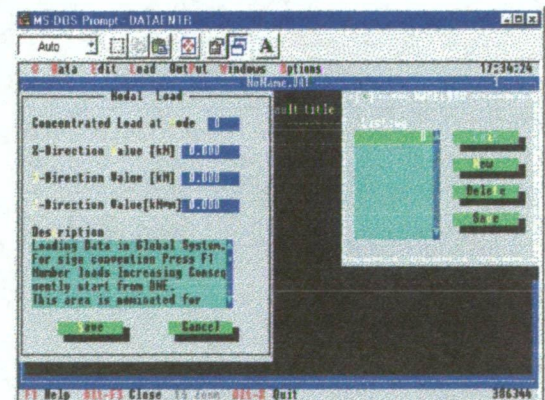


Figure A4.24 Details of Node loads applicable to structure

- “*Member Point Load MPL*” provides access to data regarding concentrated loads applicable to members. This includes assigned number to each point load; number



of member; and location of acting point along the member in *meters*. Also, amount of these loads in *X*, *Y* directions in *kN* and about *Z* direction in *kN.m*, and some space for individual technical description are included (Figures A4.25 and A4.26). Multiples of this type of load can be applied to any member.

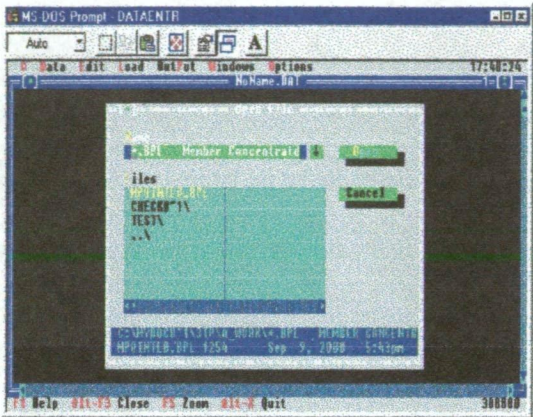


Figure A4.25 Allocating file associated with member concentrated load

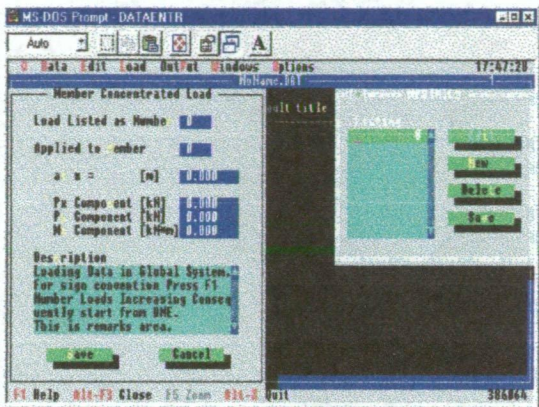


Figure A4.26 Details of member concentrated load

- “*Member Distributed Load*” with sub menu “*Uniform Load MUL*” provides access to the data regarding uniformly distributed loads applicable to members. These include the number of the member where the load applies, the amounts of the loads in *X* and *Y* directions in *kN*, and some space for individual technical notes (Figures A4.27 and A4.28).

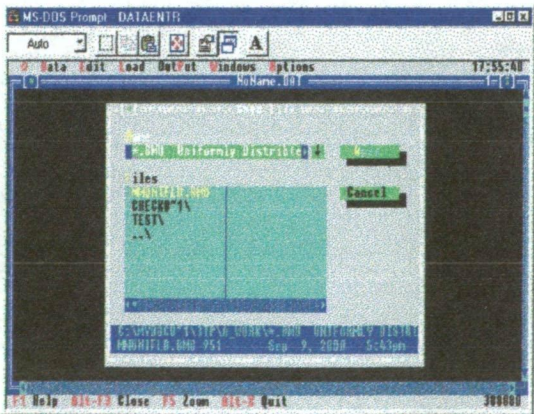


Figure A4.27 Allocating file associated with member uniformly distributed loads

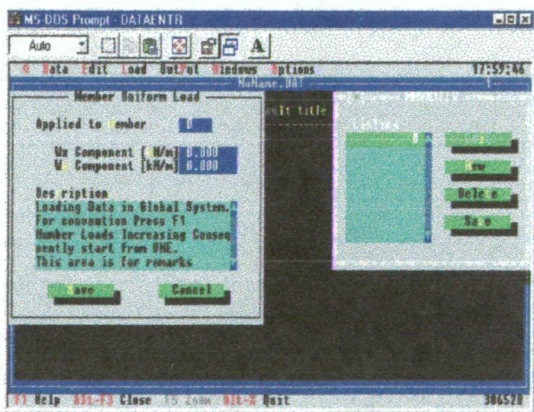


Figure A4.28 Details of member uniformly distributed loads



- “Member Trapezoidal Load MTL” under “Member Distributed Load” menu provides access to data regarding trapezoidal distributed loads that apply to structure members. This includes an assigned number to each load and member number. In addition, this programme accepts data in global or local coordinates, for location of the start point along the member (distance in *meters*) as well as the amount of force components (in  $kN/m$ ) in  $X$ , and  $Y$  directions and moment component ( $kN.m/m$ ) about  $Z$  direction. Similar space for data associated with the end point of the load as well as some space for individual technical description regarding that load is provided (Figures A4.29 and A4.30). Application of multiples of this load type to structure member is allowed.

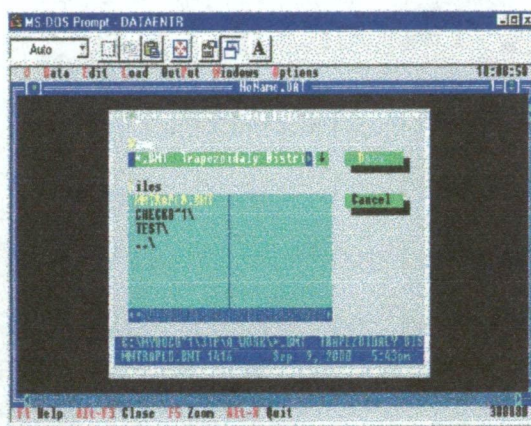


Figure A4.29 Allocation of file associated with trapezoidal loads

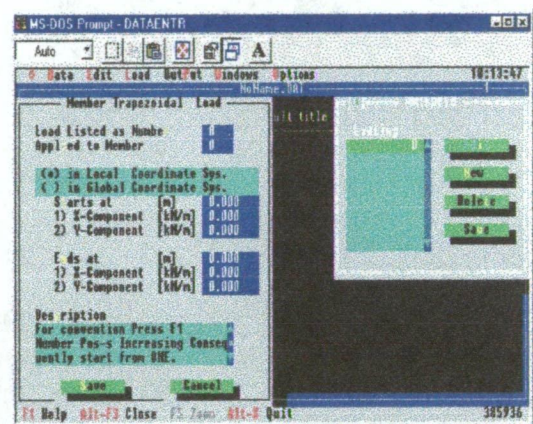


Figure A4.30 Details of trapezoidal loads applicable to structure

# Appendix A4.5

## Computer Program Listings for:

- Program GENFORMS 12 - 13
- Unit GLOBALS 14 - 20
- Program DATAENTR 21 - 43
- Program ANALYSIS 44 - 63
- Unit BUILD 64 - 79
- Unit SOILMODE 80 - 88

```

@ECHO OFF

rem next line sets up the dos and turbo compiler environment available
PATH=c:\;c:\windows;c:\windows\command;c:\progra~1\tp\bin;DOSKEY;SMARTDRV;
prompt $p$g
cd c:\mydocu~1\3tp\work;

ECHO   Programme DataEntr needs to check its initial requirements.
pause

ECHO
ECHO                                     ~~~~~
ECHO                                     ***DataEntr***
ECHO   =====
ECHO       This section of batch file generates BLANK forms for data files
ECHO       using different settings obtained from GENEDIT#.PAS and GENLOAD#.PAS
ECHO   =====
ECHO       Attempting to generate blank data files for DATAENTR program
ECHO   if there is not any type of those forms available in this directory ...
ECHO off

IF exist GEOMETRY.BGM GOTO Next1
tpc /m /q /dEDIT1 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform
IF ERRORLEVEL 1 GOTO RuntimeError

:Next1
@ECHO   There is a copy of GEOMETRY.BGM available.
IF exist MAT_PROP.BPM GOTO Next2
tpc /m /q /dEDIT2 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform
IF ERRORLEVEL 1 GOTO RuntimeError

:Next2
@ECHO   There is a copy of MAT_PROP.BPM available.
IF exist SEC_PROP.BCS GOTO Next3
tpc /m /q /dEDIT3 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform
IF ERRORLEVEL 1 GOTO RuntimeError

:Next3
@ECHO   There is a copy of SEC_PROP.BCS available.
IF exist MEMBDATA.BMD GOTO Next4
tpc /m /q /dEDIT4 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform
IF ERRORLEVEL 1 GOTO RuntimeError

:Next4
@ECHO   There is a copy of MEMBDATA.BMD available.
IF exist RESTRAIN.BRS GOTO Next5
tpc /m /q /dEDIT5 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform
IF ERRORLEVEL 1 GOTO RuntimeError

:Next5
@ECHO   There is a copy of RESTRAIN.BRS available.
IF exist PRESLOAD.BPD GOTO Next6
tpc /m /q /dLOAD1 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform
IF ERRORLEVEL 1 GOTO RuntimeError

:Next6
@ECHO   There is a copy of PRESLOAD.BPD available.
IF exist NODELOAD.BNL GOTO Next7
tpc /m /q /dLOAD2 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform
IF ERRORLEVEL 1 GOTO RuntimeError

:Next7
@ECHO   There is a copy of NODELOAD.BNL available.
IF exist MPOINTLD.BPL GOTO Next8

```



```

tpc /m /q /dLOAD3 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform
IF ERRORLEVEL 1 GOTO RuntimeError

:Next8
@ECHO   There is a copy of MPOINTLD.BPL available.
IF exist MMUNIFLD.BMU GOTO Next9
tpc /m /q /dLOAD4 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform
IF ERRORLEVEL 1 GOTO RuntimeError

:Next9
@ECHO   There is a copy of MMUNIFLD.BMU available.
IF exist MMTRAPLD.BMT GOTO EndFirstSection
tpc /m /q /dLOAD5 genform
IF ERRORLEVEL 1 GOTO CompilerError
genform

echo it is almost the last line of the first section
IF ERRORLEVEL 1 GOTO RuntimeError

goto EndFirstSection

:CompilerError
ECHO Error encountered trying to make GENFORM.PAS
GOTO EndFirstSection

:RuntimeError
ECHO Error trying to run GENFORM.EXE
GOTO EndFirstSection

:EndFirstSection

echo =====
echo   Now, the batch file (Build) compile, then executes DATAENTR program
echo   to allow user fill-in or update data in the data files.
echo =====
echo   Programme DataEntr is about to start.
echo off
pause

IF not exist DATAENTR.PAS GOTO CompileError2
tpc DataEntr /b /l
IF ERRORLEVEL 1 GOTO CompilerError2
@ECHO   DataEntr is compiled successfully. Now is about to execute.
pause

DataEntr
IF ERRORLEVEL 1 GOTO RuntimeError2

GOTO Done

:CompilerError2
Error encountered trying to build & link DataEntr.PAS
GOTO Done

:RuntimeError2
Error trying to run DataEntr.EXE
GOTO Done

:Done

```

```

Unit Globals;
(* This unit has all the variables and constants used in the *)
(* Soil And Structure Interaction Analysis Package SASIAP *)

{$N+}
INTERFACE
    (* D+,N+ Do activate when necessary*)
    USES
        Crt,Dos;

    CONST
        days : array [0..6] of String[9] =
            ('Sunday','Monday','Tuesday',
            'Wednesday','Thursday','Friday',
            'Saturday');
        Yes = ['Y','y'];
        No  = ['N','n'];
        MaxDescrLen = 250;
            (these maxima should be adjusted with the size of structure)
        MaxNodes = 6; (* 24 maximum capacity confirmed *)
        (* 25 maximum capacity confirmed after splitting the files *)
        NoFtNd = 3; (%)
    (number of nodes located for footing. this figure is the same as MaxNodes for Beams)
        MaxM_T = 3;
        MaxS_T = 3;
        MaxMembers = 8;
        MaxNdLd = 8;
        MaxPtLd = 8;
        MaxUDLd = 8;
        MaxFrame = 5; (Initial estimate for 3-D Extension model. It can be more)

        Pi = 3.14159265359;
    (INPUT DATA FILES:)
        InputFileNodeGeom = 'GEOMETRY.TGM'; {1}
        InputFileMatProp = 'MAT_PROP.TPM'; {2}
        InputFileSecProp = 'SEC_PROP.TCS'; {3}
        InputFileMemberData = 'MEMBDATA.TMD'; {4}
        InputFileRestrains = 'RESTRAIN.TRS'; {5}
        InputFilePresLoad = 'PRESLOAD.TPD'; {6}
        InputFileNodeLoad = 'NODELOAD.TNL'; {7}
        InputFileMpointLd = 'MPOINTLD.TPL'; {8}
        InputFileMMunifLd = 'MMUNIFLD.TMU'; {9}
        InputSoilModeData = 'SoilData.DAT'; {10}
        SpaceData = '3DSpace.txt'; {data2 spacing setup for neighbouring frames}
        SupportData = 'SprtDat.txt'; {data5 support setup}
        (see details regarding the format of data and files)

    (OUTPUT DATA FILES below:)
        (OutFileStructure='c:\My Documents\3tp\A_Work\*Out*.txt';)
        (data1 is the user defined OUTPUTFILE)
        OutFulCompar = 'c:\MyDocu~1\3tp\A_Work\AllOuts\FulComp.out';
    { OutFulCompar = 'c:\My Documents\3tp\A_Work\FulComp.out'; check for validity of
    path format)
        (data3 OUTPUT FILE){CMP-FD.TXT'; {TEMPORARY CHANGE NOMINATION}
        NodalG_DeflCompar = 'c:\MyDocu~1\3tp\A_Work\AllOuts\DflComp.out';
    { NodalG_DeflCompar = 'c:\My Documents\3tp\A_Work\DflComp.out'; check for validity of
    path format)
        (data4 OUTPUT FILE){CMP-D.TXT'; {TEMPORARY CHANGE NOMINATION}

    (ISOTROPIC SOIL CONSTANTS in analytical approach)
        E_Soil = 30000;(KPa) {Poisson's ratio}
        NuSoil = 0.49; {Young modulus}

    (CROSS-ANISOTROPIC SOIL CONSTANTS in numerical approach)
        Vhv = 0.49;
        Vh = 0.49;
        Vvh = 0.49;
        Eh = 30000; {Ev*Vhv/Vvh} {KPa} {Non-integer format, dependant
parameter}{2.0E4; {iso}
        Ev = 30000; {Eh*Vvh/Vhv} {KPa}
        Fv = 21429; {KPa} {Non-integer format, Fv=E/(1+V) for isotropic case}
    (ISOTROPIC SOIL CONSTANTS in numerical approach)
    (For summarised form of the numerical technique)
    { these are declared as variables but not constants due to required changes}
    { PR = 0.30; {Poisson's ratio}
    { YM = 2E4; {KPa} {young modulus}

```

```

(   Eh=EV=E=YM=E_Soil    parameters from csio. to iso.
    Vh=Vhv=VVH=V=PR=NuSoil )

TYPE
Description = STRING[MaxDescrLen];
NodeType = ARRAY[1..MaxNodes] OF RECORD
    Name      : LONGINT;
    Xcoord    : extended;
    Ycoord    : extended;
    Descr     : Description;
end;(*RECORD*)
MatType = ARRAY[1..MaxM_T] OF RECORD
    Name      : LONGINT;
    Elasticity_M: extended;
    Thermal_M  : extended;
    Descr     : Description;
end;(*RECORD*)
SecType = ARRAY[1..MaxS_T] OF RECORD
    Name      : LONGINT;
    Area      : extended;
    M_Inertia : extended;
    Width     : extended;
    Descr     : Description;
end;(*RECORD*)
MemberType = ARRAY[1..MaxMembers] OF RECORD
    Name      : LONGINT;
    S_Node    : LONGINT;
    S_Join    : LONGINT;
    E_Node    : LONGINT;
    E_Join    : LONGINT;
    C_Sec     : LONGINT;
    Mat       : LONGINT;
    L         : extended;
    Fi        : extended;
    Descr     : Description;
end;(*RECORD*)
NdRsType = ARRAY[1..MaxNodes] OF RECORD
    Name      : LONGINT;
    Xdir      : LONGINT;
    SprX      : extended;
    Ydir      : LONGINT;
    SprY      : extended;
    Zrot      : LONGINT;
    SprZ      : extended;
    Descr     : Description;
end;(*RECORD*)
P_D_LdType = ARRAY[1..MaxNodes] OF RECORD( prescribed Displ Load )
    Name      : LONGINT;
    XDispl    : EXTENDED;
    YDispl    : EXTENDED;
    ZDispl    : EXTENDED;
    XDispl_Loc : EXTENDED;
    YDispl_Loc : EXTENDED;
    Descr     : Description;
end;(*RECORD*)
NdLdType = ARRAY[1..MaxNodes] OF RECORD
    Pos      : LONGINT;
    Name      : LONGINT;
    XForce    : EXTENDED;
    YForce    : EXTENDED;
    ZMoment   : EXTENDED;
    Descr     : Description;
end;(*RECORD*)
PtLdType = ARRAY[1..MaxPtLd] OF RECORD { point Load }
    Pos      : LONGINT;
    Mem       : LONGINT;
    XLoc      : EXTENDED;
    Fx        : EXTENDED;
    Fy        : EXTENDED;
    Mz        : EXTENDED;
    Fx_L      : EXTENDED;
    Fy_L      : EXTENDED;
    Descr     : Description;
END;(*RECORD*)
UDistLdType = ARRAY[1..MaxUDLd] OF RECORD
    Mem       : LONGINT; {Uniformly Distributed Member Load}
    Ux        : EXTENDED;

```

```

    Uy          : EXTENDED;
    Ux_L        : EXTENDED;
    Uy_L        : EXTENDED;
    Descr       : Description;
END;(*RECORD*)
FtNdType = ARRAY[1..NoFtNd] OF RECORD(%)
{probably it may need to change the size of matrix to "MaxNodes"}
    Order      : LONGINT;
    Name       : LONGINT;
END;(*RECORD*)
SoilMemType = ARRAY[1..NoFtNd-1] OF RECORD(%)
    Name       : LONGINT;
    Sn         : LONGINT;
    Sx         : EXTENDED;
    En         : LONGINT;
    Ex         : EXTENDED;
    Length     : EXTENDED;
END;(*RECORD*)
PositionType = ARRAY[1..3*MaxNodes] OF RECORD
    GlobStif   : LONGINT;
    FulPopSoil : LONGINT;
end;(*RECORD*)
FrameSetUpType = ARRAY[1..MaxFrame] OF RECORD
    FramNumber : LONGINT;
    RelDistanc : EXTENDED;
    LoadFactor : EXTENDED;
end;(*RECORD*)

OneDArrayM = ARRAY[1..MaxMembers] OF EXTENDED;
Matrix3X3 = ARRAY[1..3,1..3] OF EXTENDED;
Matrix6X6 = ARRAY[1..6,1..6] OF EXTENDED;
OneDMatrixM = ARRAY[1..MaxMembers] OF Matrix6X6;
Vector6 = ARRAY[1..6] OF EXTENDED;
OneDMatrixV6 = ARRAY[1..MaxMembers] OF Vector6;
OneDMatrixN = ARRAY[1..3*MaxNodes] OF EXTENDED;
OneDArrayN = ARRAY[1..3*MaxNodes] OF LONGINT;
glnpbynp = ARRAY[1..3*MaxNodes,1..3*MaxNodes] OF EXTENDED;
glnpbymp = ARRAY[1..3*MaxNodes,1..1] OF EXTENDED;
glnp     = ARRAY[1..3*MaxNodes] OF LONGINT;
TwoDMatrix = ARRAY[1..3*MaxNodes,1..3*MaxNodes] OF EXTENDED;
OneDMatrixF = ARRAY[1..NoFtNd] OF Matrix3X3;

TwoDMatrixF = ARRAY[1..3*NoFtNd,1..3*NoFtNd] OF EXTENDED;

VAR
    Node          : NodeType;
    Mat            : MatType;
    Sec            : SecType;
    Member         : MemberType;
    NdRs           : NdRsType;
    P_D_Ld        : P_D_LdType;
    NdLd          : NdLdType;
    PtLd          : PtLdType;
    UDistLd       : UDistLdType;
    FtNd          : FtNdType;
    SoilMem        : SoilMemType;
    SoilPosDOR : PositionType;{position of soil restrains in the globstifmatrix}
    y, mont, day, dow : Word;
    h, mint, s, hund : Word;
    ConstantsAccepted,
    DOFacAccepted,
    SoilModelApplied,
    NodeOnSoil,
    ExtendedModelApplies,
    ConsiderAlone,
    ShowMeReminerNote : Char;
    data,
    data1,
    data2,
    data3,
    data4,
    data5          : TEXT;
    Emodul,        (* elastic modulus (Pa) *)
    Inr,           (* moment of inertia (m^4) *)
    L,             (* length (m) *)
    Area,          (* area(m^2) *)
    Fi             : OneDArrayM;

```

```

count,
NoNodes,
NoMats,
NoSecs,
NoMems,
NoNdRs,
NoPresLd,
NoNdLd,
NoPtLd,
NoUDistLd,
dof,           {structure kinematic degrees of freedom}
fd_dof,        {counter for k.degree of free displacement }
pd_dof,        {counter for k.degree of presc. displacement}
S_J, E_J,      {start join / end join}
EHS_dor,       {Elastic-Half-Space soil model degree of restrain}
Try            {the number of trails for Tension test in soil reaction}
              : LongInt;
MatPos,        {coefficient position in the matrix 1 to 9}
SelectedSoilModel
{0=non; 1=isotropic(ANALIT); 2=crossanisotropic(NUM); 3=isotropic(NUM)}
              : Integer;
{OutFileStructure is the variable for the user's main output defined file}
OutFileStructure,
JunkStr        : STRING;
JunkStr35      : STRING[35];
JunkChar       : CHAR;
NegReac        : Boolean;

CoefA,          {ANALYTICAL APPROACH ISOTROPIC PARAMETERS}
CoefB,
CoefC,
Rmin,           {footing width variable}
ActLength,
CumulDistanc   : EXTENDED;

              {CORSS-ANISOTROPIC SOIL VARIABLES}
( Eh,Fv: LongInt; {in attempt to make integer format of these non-int. quantities}
  PR,   {Poisson's ratio; special case OF CISO corresponds to "isotropic"}
  YM,   {KPa} {young modulus; special case OF CISO corresponds to "isotropic"}
  Aci,
  Bci,
  Cci,
  Dci,
  Fci,

  Alfa, AlfSq,
  Beta, BetSq,
  Gama, GamSq,
  Phi,
  Rho,

  G1, G2, G3, G4,
  H1, H2, H3, H4, H10,
  I1, I3,
  J1, J3, J11,
  S1, S3,
  T1, T3, T10   : Extended;

  BetaSqPos,
  BetaSqNeg,
  BetaZero      : Boolean;

  BakNum, ForNum : LONGINT;

  Temp,
  ElemTposTransMat, {transposed transformation matrix}
  ElemTransMat,     {transformation of Glob <=> Loc Systems}
  ElemL_StiffMat, ElemG_StiffMat : OneDMatrixM;

  GlobalStifMat, CombinSoilStrucStifMat : ^TwoDMatrix;

  ElemFxdFce_L, U_ElemFxdFce_L, ElemFxdFce_G,
  L_DisElem, G_DisElem, ReacElem : OneDMatrixV6;

  EffNodeFce, NodeDefl, NodeReac : OneDMatrixN;

```

```

MatPosDOF, PD_MatPosDOF, FD_MatPosDOF : OneDArrayN;

SoilFlexMat, NetSoilFlexMat, SoilStifMat : ^TwoDMatrixF;

BakFrame, ForFrame : FrameSetUpType;

StiffMatrixDOF, FD_StiffMatrixDOF, Coef_PD_StiffMatrixDOF: ^glnpbynp;
NodeFceDOF, PD_NodeDOF, PD_NodeFce : ^glnpbymp;

PROCEDURE    InitialiseGlobalVars;
PROCEDURE    InitialiseMatrixesMethodArrays;

IMPLEMENTATION

PROCEDURE    InitialiseGlobalVars;
VAR
  i : LONGINT;
BEGIN
  y:=0; mont:=0; day:=0; dow:=0; h:=0; mint:=0; s:=0; hund:=0;

  ConstantsAccepted := ' ';
  OutFileStructure  := ' ';
  DOFAccepted       := ' ';
  SoilModelApplied  := ' ';
  SelectedSoilModel := 0;
  ExtendedModelApplies:= ' ';
  ConsiderAlone     := ' ';
  NodeOnSoil        := ' ';
  JunkStr            := ' ';
  JunkChar           := ' ';
  JunkStr35          := ' ';
  MatPos             := 0;
  NegReac            := False;
  Rmin               := 0.40; (*initial value of the footing width/radius can be changed*)
  ActLength          := 0;
  CumulDistanc      := 0;

  YM      := E_Soil; {KPa Young Modulus}
  PR      := NuSoil;  {Poisson's ratio}
  CoefA   := 1;
  CoefB   := 1;
  CoefC   := 1;

  Aci     := 1;
  Bci     := 1;
  Cci     := 1;
  Dci     := 1;
  Fci     := 1;

  Alfa    := 0;
  AlfSq   := 0;
  Beta    := 0;
  BetSq   := 0;
  Gama    := 0;
  GamSq   := 0;
  Phi     := 0;
  Rho     := 0;

  G1       := 0;
  G2       := 0;
  G3       := 0;
  G4       := 0;

  H1       := 0;
  H2       := 0;
  H3       := 0;
  H4       := 0;
  H10      := 0;

  I1       := 0;
  I3       := 0;

  J1       := 0;
  J3       := 0;
  J11      := 0;

```

```

S1      := 0;
S3      := 0;

T1      := 0;
T3      := 0;
T10     := 0;

BetaSqPos := True;
BetaSqNeg := False;
BetaZero  := False;

count    := 0;
NoNodes  := 0;
NoMats   := 0;
NoSecs   := 0;
NoMems   := 0;
NoNdRs   := 0;

NoPresLd:= 0;
NoNdLd  := 0;
NoPtLd  := 0;
NoUDistLd:=0;

dof := 0; fd_dof := 0; pd_dof := 0;

S_J := 0; E_J := 0; EHS_dor := 0; Try := 0;

BakNum:=0; ForNum:=0;

FOR i:= 1 TO MaxNodes Do Begin
  With Node[i] do begin
    Name := 0; XCoord := 0; YCoord := 0; Descr := ''; end;
  With NdRs[i] do begin
    Name :=0; Xdir :=0; SprX :=0; Ydir:=0; SprY:=0; Zrot:=0; SprZ:=0;
    Descr:=''; end;
  With P_D_Ld[i] do begin
    Name:=0; XDispl:=0; YDispl:=0; ZDispl:=0; XDispl_Loc:=0; YDispl_Loc:=0;
    Descr:=''; end;
  With NdLd[i] do begin
    Pos:=0; Name:=0; XForce:=0; YForce:=0; ZMoment:=0; Descr:=''; end;
End;

FOR i:= 1 TO MaxMembers Do Begin
  With Member[i] do begin
    Name:=0; S_Node:=0; S_Join:=0; E_Node:=0; E_Join:=0; C_Sec:=0; Mat:=0;
    L:=0; Fi:=0; Descr:=''; end;

    Emodul[i] :=0; Inr[i] :=0; Area[i] :=0; L[i] :=0; Fi[i]:=0;
End;

FOR i:= 1 TO MaxM_T Do
  With Mat[i] do begin
    Name := 0; Elasticity_M:=0; Thermal_M:=0; Descr :=''; end;

FOR i:= 1 TO MaxS_T Do
  With Sec[i] do begin
    Name :=0; Area:=0; M_Inertia:=0; Descr:=''; end;

FOR i:= 1 TO MaxPtLd Do
  With PtLd[i] do begin
    Pos:=0; Mem:=0; XLoc:=0; Fx:=0; Fy:=0; Mz:=0; Fx_L:=0; Fy_L:=0;
    Descr:=''; end;

FOR i:= 1 TO MaxUDLd Do
  With UDistLd[i] do begin
    Mem:=0; Ux:=0; Uy:=0; Ux_L:=0; Uy_L:=0; Descr:=''; end;

FOR i:= 1 TO NoFtNd DO
  With FtNd[i] do begin
    Order :=0; Name :=0; end;

FOR i := 1 TO NoFtNd-1 DO
  With SoilMem[i] do begin
    Name :=0; Sn :=0; Sx :=0; En :=0; Ex :=0; Length :=0; end;

```

```

FOR i := 1 TO BakNum DO
  With BakFrame[i] do begin
    FramNumber := 0;
    LoadFactor := 0; end;

FOR i := 1 TO ForNum DO
  With ForFrame[i] do begin
    FramNumber := 0;
    LoadFactor := 0; end;
END; {procedure}

PROCEDURE InitialiseMatrixesMethodArrays;
VAR
  j, k, m : LONGINT;
BEGIN
  FOR j := 1 TO MaxMembers DO
    FOR k := 1 TO 6 DO
      FOR m := 1 TO 6 DO BEGIN
        Temp[j,k,m] := 0;
        ElemTposTransMat[j,k,m] := 0;
        ElemTransMat[j,k,m] := 0;
        ElemL_StiffMat[j,k,m] := 0;
        ElemG_StiffMat[j,k,m] := 0;
      END; {*FOR k,m*}
    FOR j := 1 TO 3*MaxNodes DO
      FOR k := 1 TO 3*MaxNodes DO BEGIN
        FD_StiffMatrixDOF^[j,k] := 0;
        Coef_PD_StiffMatrixDOF^[j,k] := 0;
        GlobalStifMat^[j,k] := 0;
        StiffMatrixDOF^[j,k] := 0;
        CombinSoilStrucStifMat^[j,k] := 0;
      END; {*FOR j & k*}
    FOR j := 1 TO MaxMembers DO
      FOR k := 1 TO 6 DO BEGIN
        ElemFxdFce_L[j,k] := 0;
        U_ElemFxdFce_L[j,k] := 0;
        ElemFxdFce_G[j,k] := 0;
        L_Dispelem[j,k] := 0;
        G_Dispelem[j,k] := 0;
        ReacElem[j,k] := 0;
      END; {*FOR j & k*}
    FOR j := 1 TO 3*MaxNodes DO BEGIN
      EffNodeFce[j] := 0;
      NodeFceDOF^[j,1] := 0;
      PD_NodeDOF^[j,1] := 0;
      PD_NodeFce^[j,1] := 0;
      NodeDefl[j] := 0;
      NodeReac[j] := 0;
      MatPosDOF[j] := 0;
      PD_MatPosDOF[j] := 0;
      FD_MatPosDOF[j] := 0;
    END; {*FOR j*}

    FOR j := 1 TO 3*MaxNodes DO {&}
      with SoilPosDOR[j] do begin
        GlobStif := 0;
        FulPopSoil := 0; end;

    FOR j := 1 TO 3*NoFtNd DO {&}
      FOR k := 1 TO 3*NoFtNd DO BEGIN
        SoilFlexMat^[j,k] := 0;
        NetSoilFlexMat^[j,k] := 0;
        SoilStifMat^[j,k] := 0;
      END; {*FOR j & k*}
    END; {*PROCEDURE*}

END.
}

```



```

Program DataStructure;

{$X+,S-}
{$M 16384,8192,655360}

uses
  Dos, Objects, Drivers, Memory, Views, Menus, Dialogs, StdDlg, MsgBox, App, Gadgets,
  FViewer, HelpFile, ColorSel, MouseDlg, Validate, AsciiTab
  , Calc, Calendar, RValidat, Editors, DataColl, MListDlg, MForms, Fields
  , FormCmds, Crt;

const
  cmAbout          = 1000;
  cmFOpen          = 1001;
  cmChDir          = 1002;
  cmMouse          = 1003;
  cmColors         = 1004;
  cmDosShell       = 1005;
  cmDataOpen       = 1006;
  cmDataNew        = 1007;
  cmDataExist      = 1008;
  cmUpdate         = 1009;
  cmUpdateFile     = 1010;
  cmUpdateName     = 1011;
  cmRenamer        = 1012;
  cmAsciiTab       = 1013;
  cmCalendar       = 1014;
  cmCalculator      = 1015;
  cmSaveDesktop    = 1016;
  cmRetrieveDesktop = 1017;
  cmPrevious       = 1018;
  cmNextEntry      = 1019;

  cmDataSaveAs     = 201;
  cmAnalyse        = 202;
  cmSaveAs         = 205;
  cmDataSave       = 206;
  cmPrintText      = 207;
  cmDataGen        = 208;
  cmNodeGeom       = 209;
  cmMatProperty    = 210;
  cmCrosSectSpec   = 211;
  cmMemOrient      = 212;
  cmRestraint      = 213;
  cmMemPrsDisplc   = 214;
  cmNodPointLd     = 215;
  cmMemPointLd     = 216;
  cmMemUnifLd      = 217;
  cmMemTripozLd    = 218;

  CommandSetData : TCommandSet = [cmAnalyse, cmDataSaveAs, cmSaveAs,
    cmDataSave, cmPrintText, cmDataGen, cmNodeGeom,
    cmMatProperty, cmCrosSectSpec, cmMemOrient, cmRestraint,
    cmMemPrsDisplc, cmNodPointLd, cmMemPointLd, cmMemUnifLd,
    cmMemTripozLd];

var
  FName          : pathstr;
  MyData         : TStruct;

const
  DataChange : Boolean = False;

type
  {TmyDataViewer}
  PmyDataViewer = ^TmyDataViewer;
  TmyDataViewer = object(TFileViewer)
    constructor Init(var Bounds: TRect; AHScrollBar,
      AVScrollBar: PScrollBar; var AFileName: PathStr);
    Procedure DataTable; Virtual;
    Procedure Draw; Virtual;
    Procedure HandleEvent(var Event: TEvent); Virtual;
  end;

  {TmyDataWindow }
  PmyDataWindow = ^TmyDataWindow;
  TmyDataWindow = object(TFileWindow)

```

```

    Constructor Init(var FName: PathStr);
    Procedure Handleevent(var Event: Tevent);           Virtual;
end;

{TmyFileViewer}
PmyFileViewer = ^TmyFileViewer;
TmyFileViewer = object(TFileViewer)
    Procedure HandleEvent(var Event: Tevent);           Virtual;
    Procedure SetState(Astate:word;Enable:Boolean);     Virtual;
end;

{TmyFileWindow }
PmyFileWindow = ^TmyFileWindow;
TmyFileWindow = object(TFileWindow)
    Constructor Init(var FileName: PathStr);
    Procedure HandleEvent(var Event: Tevent);           Virtual;
end;

{ TMyStruture }
PMyStructure = ^TMyStructure;
TMyStructure = object(TApplication)
    Clock: PClockView;
    Heap: PHeapView;
    Constructor Init;
    Procedure FileOpen(WildCard: PathStr);
    Procedure DataOpen(WildCard:PathStr);
    Procedure DataNew(WildCard : PathStr);
    Procedure DataSaveAs;
    Procedure DataSave;
    Procedure DataGen;
    Procedure NodeGeom;
    Procedure MatProperty;
    Procedure CrossSectSpec;
    Procedure MemOrient;
    Procedure Restraint;
    Procedure MemPrsDisplc;
    Procedure NodPointLd;
    Procedure MemPointLd;
    Procedure MemUnifLd;
    Procedure MemTripozLd;
    Procedure PrintText(WildCard : PathStr);
    Procedure GetEvent(var Event: TEvent);               virtual;
    Function GetPalette: PPalette;                       virtual;
    Procedure HandleEvent(var Event: TEvent);             virtual;
    Procedure Idle;                                       virtual;
    Procedure InitMenuBar;                               virtual;
    Procedure InitStatusLine;                           virtual;
    Procedure OutOfMemory;                               virtual;
    Procedure ViewData(FileName: PathStr);
    Procedure ViewFile(FileName: PathStr);
    PROCEDURE LoadDesktop(VAR S: TStream);
    PROCEDURE StoreDesktop(VAR S: TStream);
end;

{TMyStaticText}
PMyStaticText = ^TMyStaticText;
TMyStaticText = object(TStaticText)
    function GetPalette : PPalette; virtual;
end;

{ CalcHelpName }
Function CalcHelpName: PathStr;
var
    EXENAME: PathStr;
    Dir: DirStr;
    Name: NameStr;
    Ext: ExtStr;
begin
    if Lo(DosVersion) >= 3 then EXENAME := ParamStr(0)
    else EXENAME := FSearch('Structur.EXE', GetEnv('PATH'));
    FSPLIT(EXENAME, Dir, Name, Ext);
    if Dir[Length(Dir)] = '\' then Dec(Dir[0]);
    CalcHelpName := FSearch('ShHelp.HLP', Dir);
end;

{TMyStaticText}

```

```

{-----}
function TMyStaticText.GetPalette : PPalette;

const
  CMyStaticText = #1;
  PMyStaticText : string[Length(cMyStaticText)]=CMyStaticText;
begin
  GetPalette :=@PMyStaticText;
end;

(TmyDataViewer)
constructor TmyDataViewer.Init(var Bounds: TRect; AHScrollBar,
  AVScrollBar: PScrollBar; var AFileName: PathStr);
begin
  TScrollbar.Init(Bounds, AHScrollbar, AVScrollBar);
  GrowMode := gfGrowHiX + gfGrowHiY;
  FileName := nil;
  DataTable;
end;

Procedure TmyDataViewer.DataTable;

var
  Line      : String;
  MaxWidth  : Integer;
  E         : TEvent;
  S         : String;
  I         : Integer;
  J         : Integer;
  CountSegs : Integer;
  LengthSegments: ArraySeg;

Procedure SegmentList(Var J :Integer; Var Line :String);

Var
  S : String;
begin
  Str(J,S);
  Line := Line + ' ' + S;
  Str(LengthSegments[J]:2:3,S);
  if J<10 then Line := Line + ' ' +S
    else Line := Line + ' ' +S;
  inc(J);
end;
begin
  MyData.GetlenSegment(LengthSegments);
  CountSegs := MyData.GetSegsBeloDL + MyData.GetSegsAbovDL;
  I := 1;
  isvalid:=true;
  FileLines := New(PLineCollection, Init(5,5));
  MaxWidth := 0;
  while (I < 47) and not LowMemory do
    begin
      case I of
        2 :   Line := ' ~ '+MyData.GetTitle;
        3 :   Line := ' ~ -----';
        else Line :='';
      end;
      if Length(Line) > MaxWidth then MaxWidth := Length(Line);
      FileLines^.Insert(NewStr(Line));
      Inc(I);
    end;

    Limit.X := MaxWidth;
    Limit.Y := FileLines^.Count;
  end;

Procedure TmyDataViewer.Draw;

Var
  B: TDrawBuffer;
  I: Integer;
  S: String;
  P: PString;

begin

```

```

for I := 0 to Size.Y - 1 do
begin
  MoveChar(B, ' ', $02, size.X);

  if Delta.Y + I < FileLines^.Count then
  begin
    P := FileLines^.At(Delta.Y + I);
    if P <> nil then S := Copy(P^, Delta.X + 1, Size.X)
    else S := '';
    MovecStr(B, S, $0E02)
  end;
  WriteLine(0, I, Size.X, 1, B);
end;
end;

Procedure TmyDataViewer.HandleEvent(var Event:TEvent);

begin
  TfileViewer.HandleEvent(Event);
  if(Event.what = evBroadcast) and (event.command = cmUpdate) then
  begin
    ClearEvent(event);
    Filelines^.done;
    Drawview;
  end;
end;

{TmyDataWindow}

Constructor TmyDataWindow.Init(var FName: PathStr);

const
  WinNumber: Integer = 1;
var
  R: TRect;
begin
  Desktop^.GetExtent(R);
  TWindow.Init(R, FName, WinNumber);
  Options := Options or ofTileable;
  Inc(WinNumber);
  GetExtent(R);
  R.Grow(-1, -1);
  Insert(New(PmyDataViewer, Init(R,
    StandardScrollBar(sbHorizontal + sbHandleKeyboard),
    StandardScrollBar(sbVertical + sbHandleKeyboard), FName)));
end;

Procedure TmyDataWindow.Handleevent(var Event: tevent);
var
  control : word;
begin
  if(Event.what=evcommand) then
  begin
    begin
      case event.command of
        cmclose : begin
          if DataChange then
          begin
            Control := MessageBox(' Data has changed - Save changes ?',
              nil,mfwarning+ mfliesButton+mfNoButton);
            if control = cmyes then
              message(Desktop^.owner, evcommand, cmdatasave, nil);
            end;
            Close;
            ClearEvent(event);
          end;
        cmquit : begin
          if DataChange then
          begin
            zoom;
            Control := MessageBox(' Data has changed - Save changes ?',
              nil,mfwarning+ mfliesButton+mfNoButton);
            if control = cmyes then
              message(Desktop^.owner, evcommand, cmdatasave, nil);
            end;
            endmodal(cmquit);
            ClearEvent(event);
          end;
        end;
      end;
    end;
  end;
end;

```

```

        end;
    end;
    TfileWindow.HandleEvent(Event);
    if(Event.what = evBroadcast) then
    begin
        case event.command of
            cmDataExist :   ClearEvent(event);
        end;
    end;
end;

{TmyFileViewer}

Procedure TmyFileViewer.SetState(Astate:word; enable:Boolean);

begin
    TFileViewer.SetState(Astate,enable);
    if Astate and sfactive <>0 then
        if enable then enableCommands([cmSaveAs])
        else disableCommands([cmSaveAs]);
    end;

Procedure TmyFileViewer.Handleevent(var Event: tevent);

Procedure Save(Filename:Pstring);

var
    FileToView: Text;
    I          : Integer;
    P          : PString;
    S          : String;

begin
    {$I-}
    IsValid := True;
    if FileName <> nil then DisposeStr(FileName);
    FileName := NewStr(FName);
    Assign(FileToView, FName);
    Rewrite(FileToView);
    if IOResult <> 0 then
        begin
            MessageBox('Cannot save file '+FName+'.', nil, mfError + mfOkButton);
            IsValid := False;
        end
    else
        begin
            for I := 0 to FileLines^.Count-1 do
                begin
                    P := FileLines^.At(I);
                    if P <> nil then S := Copy(P^, 1, limit.X)
                    else S := '';
                    WriteLn(filetoview,S);
                end;
            end;
        end;
    {$I+}
    end;
    close(filetoview);
end;

begin
    TFileViewer.HandleEvent(Event);
    if(Event.what = evBroadcast) and (event.command = cmUpdateFile) then
    begin
        Save(Filename);
        ClearEvent(event);
    end;
end;

{TmyFileWindow}
constructor TmyFileWindow.Init(var FileName: PathStr);
const
    WinNumber: Integer = 1;
var
    R: TRect;
begin
    Desktop^.GetExtent(R);
    TWindow.Init(R, FileName, WinNumber);

```

```

Options := Options or ofTileable;
Inc(WinNumber);
GetExtent(R);
R.Grow(-1, -1);
Insert(New(PmyFileViewer, Init(R,
  StandardScrollBar(sbHorizontal + sbHandleKeyboard),
  StandardScrollBar(sbVertical + sbHandleKeyboard), Filename)));
end;

Procedure TmyFileWindow.HandleEvent(var Event: tevent);
var
  DataExists : PMyDataWindow;

Begin
  if (Event.what=evcommand) and (Event.command=cmquit) then
    begin
      DataExists :=Message(Desktop,evBroadcast,cmDataExist,nil);
      if DataExists <> nil then
        begin
          DataExists^.Select;
          DataExists^.Zoom;
          DataExists^.HandleEvent(event);
        end;
      end;
    TfileWindow.HandleEvent(Event);
    if(Event.what = evBroadcast) and (event.command = cmUpdateName) then
      begin
        Title^:=Fname;
        Frame^.DrawView;
        ClearEvent(event);
      end;
    end;
end;

{ TMyStructure }
Constructor TMyStructure.Init;
var
  R      : TRect;
  I      : Integer;
  FileName: PathStr;
  Event  : TEvent;
begin
  TApplication.Init;
  RegisterObjects;
  RegisterViews;
  RegisterMenus;
  RegisterCalendar;
  RegisterAsciiTab;
  RegisterCalc;
  RegisterDialogs;
  RegisterApp;
  RegisterHelpFile;
  RegisterFViewer;
  RegisterDataColl;
  RegisterForms;
  RegisterEditors;
  RegisterFields;
  RegisterType(RStruct);
  RegisterValidate;
  RegisterMyValidators;
  myData.init;

  GetExtent(R);
  R.A.X := R.B.X - 9; R.B.Y := R.A.Y+1;
  Clock := New(PClockView, Init(R));
  Insert(Clock);

  GetExtent(R);
  Dec(R.B.X);
  R.A.X := R.B.X - 9; R.A.Y := R.B.Y - 1;
  Heap := New(PHeapView, Init(R));
  Insert(Heap);

  {Display About Box}
  Event.What := evCommand;

```

```

Event.Command :=cmAbout;
PutEvent(Event);

Disablecommands([cmSaveas]);

for I := 1 to ParamCount do
begin
    FileName := ParamStr(I);
    if FileName[Length(FileName)] = '\' then
        FileName := FileName + '*..*';
    if (Pos('?', FileName) = 0) and (Pos('*', FileName) = 0) then
        ViewFile(FExpand(FileName))
    else FileOpen(FileName);
end;
end;

Procedure TMyStructure.FileOpen(WildCard: PathStr);

var
    D      : PFileDialog;
    FileName: PathStr;

begin
    D := New(PFileDialog, Init(WildCard, 'Open a File',
        '~N-ame', fdOpenButton + fdHelpButton, 100));
    D^.HelpCtx := hcFOFileOpenDBox;
    if ValidView(D) <> nil then
        begin
            if Desktop^.ExecView(D) <> cmCancel then
                begin
                    D^.GetFileName(FileName);
                    ViewFile(FileName);
                end;
            Dispose(D, Done);
        end;
    end;

Procedure TMyStructure.DataOpen(WildCard:PathStr);

Var
    OpenBox      : PfileDialog;
    FCStream      : TBufStream;
    Control       : word;
    StructData    : PStruct;
    DataExists    : PMyDataWindow;

Begin
    DataExists :=Message(Desktop,evBroadcast,cmDataExist,nil);
    if DataExists <> nil then
        begin
            DataExists^.Select;
            Message (TopView,evcommand,cmclose,nil);
        end;
    OpenBox:= New(PFileDialog, Init('*.*.dat','Open Data File',
        'Data File Name',fdokbutton + fdOpenButton,1));
    OpenBox^.HelpCtx := hcFOFileOpenDBox;
    if ValidView(OpenBox) <> nil then
        begin
            if Desktop^.ExecView (OpenBox) <> cmCancel then
                begin
                    OpenBox^.GetFilename(Fname);
                    FCStream.Init(Fname,stOpenRead,512);
                    StructData := PStruct(FCStream.Get);
                    if FCStream.status <> stok then
                        MessageBox('File error - File wrong type?',
                            nil, mfError + mfOkButton)
                    else
                        begin
                            MyData := StructData^;
                            Dispose(StructData);
                            ViewData(Fname);
                            DataChange := False;
                        end;
                end;
            end;

```





```

    if Desktop^.ExecView(SaveBox) <> cmCancel then
    begin
        SaveBox^.GetFileName(fname);
        FCStream.Init(Fname,stCreate,512);
        FCStream.Put(@MyData);
        if FCStream.status <> stok then
            MessageBox(
                'File not saved - read only?    Use another filename',
                nil, mfError + mfOkButton)
            else
            begin
                DataChange := False;
                Message (TopView,evcommand,cmclose,nil);
                ViewData(FName);
            end;
            FCStream.Done;
        end;
        dispose(SaveBox, done);
    end;
end;

Procedure TmyStructure.DataGen;

Type
    TEditData = Record
        Title   : TitleStr;
        ListBox : word;
    end;

Var P      : Pdialog;
    R      : TRect;
    Control : word;
    EditData : TeditData;
    S       : string;
    A,B,C,D : Pview;

Begin
    with EditData do
    begin
        Title:=MyData.GetTitle;
    end;
    R.Assign(15,3,52,15);
    P := New(PDialog,Init(r,'Edit General Data'));
    P^.HelpCtx := hcDGeneral;
    if lowmemory then
    begin
        OutofMemory;
        Control := cmcancel;
    end
    else
    begin
        if validview (P) <> nil then
        begin
            with P^ do
            begin
                R.assign (3,5,Size.x-4,6);
                A:= New(pInputLine,Init(R,TitleMaxLen));
                Insert (A);
                R.assign (3,2,30,3);
                Insert(New(plabel,init(R,'General Description of Job',A)));
                R.assign ( 20,9,32,11);
                Insert(New(Pbutton,Init(r,'Cancel',cmcancel,bfnormal)));
                R.assign (5,9,17,11);
                Insert(New(Pbutton,Init(r,'~O~k',cmOk,bfdefault)));
                SetData(EditData);
            end;
            Control :=Desktop^.ExecView(P);
            If Control<> cmCancel then
            begin
                P^.GetData(EditData);
                DataChange := True;
                MyData.SetTitle(EditData.Title);
                Message(Desktop^.owner,evBroadcast,cmUpDate,Nil);
                Dispose(P,done);
            end
            else
                dispose(P,Done);
            end;

```

```

        end;
    end;
end;

PROCEDURE TmyStructure.NodeGeom;
var
    D: PFileDialog;
    FileName: ^PathStr;
    ListEditor: PDialog;
begin
    D := New(PFileDialog, Init('*.BGM Node Geometry', 'Open File',
        '~N-ame', fdOpenButton, hOpenListDlg));
    if ValidView(D) <> nil then
    begin
        if Desktop^.ExecView(D) <> cmCancel then
        begin
            New(FileName);
            D^.GetFileName(FileName^);
            if not FileExists(FileName^) then
            begin
                MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton);
            end
            else
            begin
                { If ListEditor exists, select it; otherwise, open new one *****}
                ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
                if ListEditor = nil then
                    Desktop^.Insert(ValidView(New(PListDialog, Init(FileName^))))
                else ListEditor^.Select;
            end;
            Dispose(FileName);
        end;
        Dispose(D, Done);
    end;
end;

{writeln(FName, ' is the name of the main file. ');readln;}

end;

(*PROCEDURE NodeGeom*)

Procedure TmyStructure.MatProperty;
var
    D: PFileDialog;
    FileName: ^PathStr;
    ListEditor: PDialog;
begin
    D := New(PFileDialog, Init('*.BPM Material Properties', 'Open File',
        '~N-ame', fdOpenButton, hOpenListDlg));
    if ValidView(D) <> nil then
    begin
        if Desktop^.ExecView(D) <> cmCancel then
        begin
            New(FileName);
            D^.GetFileName(FileName^);
            if not FileExists(FileName^) then
                MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton)
            else
            begin
                { If ListEditor exists, select it; otherwise, open new one *****}
                ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
                if ListEditor = nil then
                    Desktop^.Insert(ValidView(New(PListDialog, Init(FileName^))))
                else ListEditor^.Select;
            end;
            Dispose(FileName);
        end;
        Dispose(D, Done);
    end;
end;

end;

Procedure TmyStructure.CrossSectSpec;
var
    D: PFileDialog;
    FileName: ^PathStr;
    ListEditor: PDialog;

```

```

begin
  D := New(PFileDialog, Init('*..BCS Cross Section Properties', 'Open File',
    '~N~ame', fdOpenButton, hlOpenListDlg));
  if ValidView(D) <> nil then
  begin
    if Desktop^.ExecView(D) <> cmCancel then
    begin
      New(FileName);
      D^.GetFileName(FileName^);
      if not FileExists(FileName^) then
        MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton)
      else
      begin
        { If ListEditor exists, select it; otherwise, open new one *****}
        ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
        if ListEditor = nil then
          Desktop^.Insert(ValidView(New(PListDialog, Init(FileName^))))
        else ListEditor^.Select;
      end;
      Dispose(FileName);
    end;
    Dispose(D, Done);
  end;
end;

Procedure TmyStructure.MemOrient;
var
  D: PFileDialog;
  FileName: ^PathStr;
  ListEditor: PDialog;
begin
  D := New(PFileDialog, Init('*..BMD Member Data', 'Open File',
    '~N~ame', fdOpenButton, hlOpenListDlg));
  if ValidView(D) <> nil then
  begin
    if Desktop^.ExecView(D) <> cmCancel then
    begin
      New(FileName);
      D^.GetFileName(FileName^);
      if not FileExists(FileName^) then
        MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton)
      else
      begin
        { If ListEditor exists, select it; otherwise, open new one *****}
        ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
        if ListEditor = nil then
          Desktop^.Insert(ValidView(New(PListDialog, Init(FileName^))))
        else ListEditor^.Select;
      end;
      Dispose(FileName);
    end;
    Dispose(D, Done);
  end;
end;

Procedure TmyStructure.Restraint;
var
  D: PFileDialog;
  FileName: ^PathStr;
  ListEditor: PDialog;
begin
  D := New(PFileDialog, Init('*..BRS Structure Restraints', 'Open File',
    '~N~ame', fdOpenButton, hlOpenListDlg));
  if ValidView(D) <> nil then
  begin
    if Desktop^.ExecView(D) <> cmCancel then
    begin
      New(FileName);
      D^.GetFileName(FileName^);
      if not FileExists(FileName^) then
        MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton)
      else
      begin
        { If ListEditor exists, select it; otherwise, open new one *****}
        ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
        if ListEditor = nil then
          Desktop^.Insert(ValidView(New(PListDialog, Init(FileName^))))

```

```

        else ListEditor^.Select;
    end;
    Dispose(FileName);
end;
Dispose(D, Done);
end;
end;      (*PROCEDURE Restraint*)

```

```

Procedure TmyStructure.MemPrsDisplc;

```

```

var
    D: PFileDialog;
    FileName: ^PathStr;
    ListEditor: PDialog;
begin
    D := New(PFileDialog, Init('*.BPD Prescribed Displacement', 'Open File',
        '~N-ame', fdOpenButton, h1OpenListDlg));
    if ValidView(D) <> nil then
    begin
        if Desktop^.ExecView(D) <> cmCancel then
        begin
            New(FileName);
            D^.GetFileName(FileName^);
            if not FileExists(FileName^) then
                MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton)
            else
            begin
                { If ListEditor exists, select it; otherwise, open new one *****}
                ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
                if ListEditor = nil then
                    Desktop^.Insert(ValidView(New(PListDialog, Init(FileName^))))
                else ListEditor^.Select;
            end;
            Dispose(FileName);
        end;
        Dispose(D, Done);
    end;
end;

```

```

Procedure TmyStructure.NodPointLd;

```

```

var
    D: PFileDialog;
    FileName: ^PathStr;
    ListEditor: PDialog;
begin
    D := New(PFileDialog, Init('*.BNL Nodal Loads', 'Open File',
        '~N-ame', fdOpenButton, h1OpenListDlg));
    if ValidView(D) <> nil then
    begin
        if Desktop^.ExecView(D) <> cmCancel then
        begin
            New(FileName);
            D^.GetFileName(FileName^);
            if not FileExists(FileName^) then
                MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton)
            else
            begin
                { If ListEditor exists, select it; otherwise, open new one *****}
                ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
                if ListEditor = nil then
                    Desktop^.Insert(ValidView(New(PListDialog, Init(FileName^))))
                else ListEditor^.Select;
            end;
            Dispose(FileName);
        end;
        Dispose(D, Done);
    end;
end;

```

```

Procedure TmyStructure.MemPointLd;

```

```

var
    D: PFileDialog;
    FileName: ^PathStr;
    ListEditor: PDialog;
begin
    D := New(PFileDialog, Init('*.BPL Member Concentrated', 'Open File',
        '~N-ame', fdOpenButton, h1OpenListDlg));

```

```

if ValidView(D) <> nil then
begin
  if Desktop^.ExecView(D) <> cmCancel then
  begin
    New(FileName);
    D^.GetFileName(FileName^);
    if not FileExists(FileName^) then
      MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton)
    else
      begin
        { If ListEditor exists, select it; otherwise, open new one *****}
        ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
        if ListEditor = nil then
          DeskTop^.Insert(ValidView(New(PListDialog, Init(FileName^))))
        else ListEditor^.Select;
      end;
    Dispose(FileName);
  end;
  Dispose(D, Done);
end;

end;

Procedure TmyStructure.MemUnifLd;
var
  D: PFileDialog;
  FileName: ^PathStr;
  ListEditor: PDialog;
begin
  D := New(PFileDialog, Init('*.BMU Uniformly Distributed', 'Open File',
    '~N~ame', fdOpenButton, hOpenListDlg));
  if ValidView(D) <> nil then
  begin
    if Desktop^.ExecView(D) <> cmCancel then
    begin
      New(FileName);
      D^.GetFileName(FileName^);
      if not FileExists(FileName^) then
        MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton)
      else
        begin
          { If ListEditor exists, select it; otherwise, open new one *****}
          ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
          if ListEditor = nil then
            DeskTop^.Insert(ValidView(New(PListDialog, Init(FileName^))))
          else ListEditor^.Select;
        end;
      Dispose(FileName);
    end;
  end;
  Dispose(D, Done);
end;

end;

Procedure TmyStructure.MemTripozLd;
var
  D: PFileDialog;
  FileName: ^PathStr;
  ListEditor: PDialog;
begin
  D := New(PFileDialog, Init('*.BMT Trapezoidaly Distributed', 'Open File',
    '~N~ame', fdOpenButton, hOpenListDlg));
  if ValidView(D) <> nil then
  begin
    if Desktop^.ExecView(D) <> cmCancel then
    begin
      New(FileName);
      D^.GetFileName(FileName^);
      if not FileExists(FileName^) then
        MessageBox('Cannot find file (%s).', @FileName, mfError + mfOkButton)
      else
        begin
          { If ListEditor exists, select it; otherwise, open new one *****}
          ListEditor := Message(Desktop, evBroadcast, cmEditingFile, FileName);
          if ListEditor = nil then
            DeskTop^.Insert(ValidView(New(PListDialog, Init(FileName^))))
          else ListEditor^.Select;
        end;
    end;
  end;
end;

```

```

        end;
        Dispose(FileName);
    end;
    Dispose(D, Done);
end; end;
Procedure TMyStructure.GetEvent(var Event: TEvent);

var
    W      : PWindow;
    HFile  : PHelpFile;
    HelpStrm: PDosStream;

const
    HelpInUse: Boolean = False;

begin
    TApplication.GetEvent(Event);
    case Event.What of
        evCommand:
            if (Event.Command = cmHelp) and not HelpInUse then
                begin
                    HelpInUse:= True;
                    HelpStrm := New(PDosStream, Init(CalcHelpName, stOpenRead));
                    HFile    := New(PHelpFile, Init(HelpStrm));
                    if HelpStrm^.Status <> stOk then
                        begin
                            MessageBox('Could not open help file.', nil, mfError + mfOkButton);
                            Dispose(HFile, Done);
                        end
                    else
                        begin
                            W := New(PHelpWindow, Init(HFile, GetHelpCtx));
                            if ValidView(W) <> nil then
                                begin
                                    ExecView(W);
                                    Dispose(W, Done);
                                end;
                                    ClearEvent(Event);
                                end;
                                    HelpInUse := False;
                                end;
                            evMouseDown:
                                if Event.Buttons <> 1 then Event.What := evNothing;
                            end;
                        end;
                    end;

Function TMyStructure.GetPalette: PPalette;
const
    CNewColor      = CAppColor + CHelpColor;
    CNewBlackWhite = CBlackWhite + CHelpBlackWhite;
    CNewMonochrome = CMonochrome + CHelpMonochrome;
    P: array[apColor..apMonochrome] of string[Length(CNewColor)] =
        (CNewColor, CNewBlackWhite, CNewMonochrome);
begin
    GetPalette := @P[AppPalette];
end;

Procedure TMyStructure.HandleEvent(var Event: TEvent);
var
    NewMode: Word;

Procedure OpenMemUnifLd;
var
    D: PFileDialog;
    FileName: ^PathStr;
    ListEditor: PDialog;
begin
    D := New(PFileDialog, Init('*.MUL Uniformly Distributed', 'Open File',
        '~N~ame', fdOpenButton, h1OpenListDlg));
    if ValidView(D) <> nil then
        begin
            if Desktop^.ExecView(D) <> cmCancel then
                begin
                    New(FileName);
                    D^.GetFileName(FileName^);
                    if not FileExists(FileName^) then
                        MessageBox('Cannot find file {s}.', @FileName, mfError + mfOkButton)
                end
            end
        end
    end;
end;

```



```

        ^C('Press F1 key for Help at any time'))));
    R.Assign(25, 17, 35, 19);
    Insert(New(PButton, Init(R, 'O-K', cmOk, bfDefault)));
end;
if ValidView(D) <> nil then
begin
    Desktop^.ExecView(D);
    Dispose(D, Done);
end;
end;

Procedure RetrieveDesktop;
VAR
    S: PStream;
BEGIN
    S := New(PBufStream, Init('AXS.DSK', stOpenRead, 1024));
    IF LowMemory THEN OutOfMemory
    ELSE IF S^.Status <> stOk THEN
        MessageBox('Could not open desktop file', nil, mfOkButton + mfError)
    ELSE
        BEGIN
            LoadDesktop(S^);
            IF S^.Status <> stOk THEN
                MessageBox('Error reading desktop file', nil, mfOkButton + mfError);
            END;
            Dispose(S, Done);
        END;
END;(*PROCEDURE RetrieveDesktop*)

Procedure SaveDesktop;
VAR
    S: PStream;
    F: File;
BEGIN
    S := New(PBufStream, Init('AXS.DSK', stCreate, 1024));
    IF not LowMemory and (S^.Status = stOk) THEN
        BEGIN
            StoreDesktop(S^);
            IF S^.Status <> stOk THEN
                BEGIN
                    MessageBox('Could not create AXS.DSK.', nil, mfOkButton + mfError);
                    {$I-}
                    Dispose(S, Done);
                    Assign(F, 'AXS.DSK');
                    Erase(F);
                    Exit;
                END;
            END;
            Dispose(S, Done);
        END;
END;(*PROCEDURE SaveDesktop*)

procedure Colors;
var
    D: PColorDialog;
begin
    D := New(PColorDialog, Init('',
        ColorGroup('Desktop',
            ColorItem('Color', 32, nil),
            ColorGroup('Menus',
                ColorItem('Normal', 2,
                    ColorItem('Disabled', 3,
                        ColorItem('Shortcut', 4,
                            ColorItem('Selected', 5,
                                ColorItem('Selected disabled', 6,
                                    ColorItem('Shortcut selected', 7, nil)))))),
                ColorGroup('Dialogs',
                    ColorItem('Frame/background', 33,
                        ColorItem('Frame icons', 34,
                            ColorItem('Scroll bar page', 35,
                                ColorItem('Scroll bar icons', 36,
                                    ColorItem('Static text', 37,
                                        ColorItem('Label normal', 38,
                                            ColorItem('Label selected', 39,
                                                ColorItem('Label shortcut', 40,

```



```

        ColorItem('Button normal',      41,
        ColorItem('Button default',     42,
        ColorItem('Button selected',    43,
        ColorItem('Button disabled',    44,
        ColorItem('Button shortcut',    45,
        ColorItem('Button shadow',      46,

        ColorItem('Cluster normal',     47,
        ColorItem('Cluster selected',   48,
        ColorItem('Cluster shortcut',   49,

        ColorItem('Input normal',       50,
        ColorItem('Input selected',     51,
        ColorItem('Input arrow',        52,

        ColorItem('History button',     53,
        ColorItem('History sides',      54,
        ColorItem('History bar page',   55,
        ColorItem('History bar icons',  56,

        ColorItem('List normal',        57,
        ColorItem('List focused',       58,
        ColorItem('List selected',      59,
        ColorItem('List divider',       60,

        ColorItem('Information pane',   61, nil)))))))))))))))))))))))).
    ColorGroup('Output',
        ColorItem('Frame passive',      8,
        ColorItem('Frame active',       9,
        ColorItem('Frame icons',       10,
        ColorItem('Scroll bar page',   11,
        ColorItem('Scroll bar icons',  12, nil))),

    ColorGroup('Ascii table',
        ColorItem('Frame passive',      24,
        ColorItem('Frame active',       25,
        ColorItem('Frame icons',       26,
        ColorItem('Scroll bar page',   27,
        ColorItem('Scroll bar icons',  28,
        ColorItem('Text',              29, nil))))), nil))));

D^.HelpCtx := hcOCColorsDBox;
if ValidView(D) <> nil then
begin
    D^.SetData(Application^.GetPalette^);
    if Desktop^.ExecView(D) <> cmCancel then
    begin
        Application^.GetPalette^ := D^.Pal;
        DoneMemory; { Dispose all group buffers }
        Redraw;      { Redraw application with new palette }
    end;
    Dispose(D, Done);
end;
end;

Procedure Calendar;
VAR
    P: PCalendarWindow;
BEGIN
    P := New(PCalendarWindow, Init);
    P^.HelpCtx := hcOMMousedBox;
    Desktop^.Insert(ValidView(P));
END;(*PROCEDURE Calendar*)

Procedure AsciiTab;
VAR
    P: PAsciiChart;
BEGIN
    P := New(PAsciiChart, Init);
    P^.HelpCtx := hcOMMousedBox;
    Desktop^.Insert(ValidView(P));
END;(*PROCEDURE AsciiTab*)

Procedure Calculator;
VAR
    P: PCalculator;

```

```

BEGIN
  P := New(PCalculator, Init);
  P^.HelpCtx := hcOMMousedBox;
  IF ValidView(P) <> nil THEN
    Desktop^.Insert(P);
END; (*PROCEDURE Calculator*)

Procedure Mouse;

var
  D: PDialog;

begin
  D := New(PMouseDialog, Init);
  D^.HelpCtx := hcOMMousedBox;
  if ValidView(D) <> nil then
    begin
      D^.SetData(MouseReverse);
      if Desktop^.ExecView(D) <> cmCancel then
        D^.GetData(MouseReverse);
    end;
end;

Procedure DosShell7;

begin
  DoneSysError;
  DoneEvents;
  DoneVideo;
  DoneMemory;
  SetMemTop(HeapPtr);
  PrintStr('Type EXIT to return...');
  SwapVectors;
  Exec(GetEnv('COMSPEC'), '');
  SwapVectors;
  SetMemTop(HeapEnd);
  InitMemory;
  InitVideo;
  InitEvents;
  InitSysError;
  Redraw;
end;

Procedure Analyse;

Procedure SaveRun;

Var
  FileSave :Text;
  DataExists : pMyDataWindow;
begin
  DataExists :=Message(Desktop,evBroadcast,cmDataExist,nil);
  if DataExists <>nil then DataExists^.Select;
  {$I-}
  Assign(FileSave, 'Structure.Tmp');
  Rewrite(FileSave);
  if IOResult <> 0 then
    begin
      MessageBox('Cannot save file data to run ', nil, mfError + mfOkButton);
    end
  else
    begin
      MyData.SaveData(FileSave);
    end;
  Close(FileSave);
  {$I+}
end;

Procedure ReNamer;

var
  Name : Text;
  S : PathStr;

begin
  {$I-}
  S := FName;

```

```

    S := copy(S,1,length(S)-4)+'.OUT';
    Assign(Name,S);
    Reset(Name);
    Close(Name);
    If IoResult = 0 then Erase(Name);
    Assign(Name,'Structure.OUT');
    Rename(Name,S);
    if ioResult<>0 then FName := 'Structure.OUT'
    else FName:=S;
    {$I+}
end;
begin
    SaveRun;
    if (MemAvail < 250000)then
        MessageBox('Insufficient memory to solve problem. Close applications.'
            ,nil,mfError+mfOkButton)
    else
        begin
            DoneSysError;
            DoneEvents;
            DoneVideo;
            DoneMemory;
            SetMemTop(HeapPtr);
            SwapVectors;
            Exec('C:\command.com','/c sht.bat');
            SwapVectors;
            SetMemTop(HeapEnd);
            InitMemory;
            InitVideo;
            InitEvents;
            InitSysError;
            Redraw;
            Renamer;
            ViewFile(fname);
            Message(topview,evcommand,cmtile,nil);
        end;
    end;
end;

```

Procedure SaveAs(WildCard: PathStr);

```

var
    D: PFileDialog;
    FileName: PathStr;

begin
    D := New(PFileDialog, Init(WildCard, 'Save File As',
        '~N~ame', fdOkbutton, 100));
    D^.HelpCtx:=hcFSSaveAs;
    if ValidView(D) <> nil then
        begin
            if Desktop^.ExecView(D) <> cmCancel then
                begin
                    D^.GetFileName(FileName);
                    fname:=filename;
                    Message(desktop^.owner,evBroadcast,cmUpdateName,nil);
                    Message(desktop^.owner,evBroadcast,cmUpdateFile,nil);
                end;
            Dispose(D, Done);
        end;
    end;
end;

```

end;  
 Procedure Renamer(WildCard: PathStr);

```

var
    D: PFileDialog;
    FileName      : PathStr;
    FName         : PathStr;
    Name          : text;
    FirstBoxOk    : Boolean;
    SecondBoxOk   : Boolean;

begin
    {$I-}
    FirstBoxOk    :=True;
    SecondBoxOk   :=True;

```

```

D := New(PFileDialog, Init(WildCard, 'Rename File',
    '~N~ame', fdOkbutton, 100));
if ValidView(D) <> nil then
begin
    if Desktop^.ExecView(D) <> cmCancel then
        D^.GetFileName(FileName)
        else FirstBoxOk:=False;
        Dispose(D, Done);
    end;
if FirstBoxOk then
begin
    D := New(PFileDialog, Init(WildCard, 'New File Name',
        '~N~ame', fdOkbutton, 100));
    if ValidView(D) <> nil then
        begin
            if Desktop^.ExecView(D) <> cmCancel then
                D^.GetFileName(FReName)
                else SecondBoxOk :=False;
                Dispose(D, Done);
            end;
            if SecondBoxOk then
                begin
                    Assign (Name,Filename);
                    if Filename <> FReName then Rename(Name,FReName);
                end;
            if IOResult <> 0 then
                begin
                    MessageBox('Cannot rename file '+FileName+'.', nil, mfError + mfOkButton);
                end;
            end;
            ($I+)
        end;
end;

PROCEDURE VideoMode;
begin
    NewMode := ScreenMode xor smFont8x8;
    if NewMode and smFont8x8 <> 0 then
        ShadowSize.X := 1
    else ShadowSize.X := 2;
        SetScreenMode(NewMode);
end;(*PROCEDURE VideoMode;*)

begin
    TApplication.HandleEvent(Event);
    case Event.What of
        evCommand:
            begin
                case Event.Command of
                    cmFOpen      : FileOpen('*.OUT');
                    cmSaveAs     : SaveAs('*.');
                    cmRenamer    : Renamer('*.');
                    cmDataOpen   : DataOpen('*.DAT');
                    cmDataNew    : DataNew('NoName.DAT');
                    cmDataSave   : DataSave;
                    cmDataSaveAs : DataSaveAs;
                    cmPrintText  : PrintText('*.dat');
                    cmDataGen    : DataGen;
                    cmNodeGeom   : NodeGeom;
                    cmMatProperty : MatProperty;
                    cmCrossSectSpec : CrossSectSpec;
                    cmMemOrient  : MemOrient;
                    cmRestraint  : Restraint;
                    cmMemPrsDisplc : MemPrsDisplc;
                    cmNodPointLd : NodPointLd;
                    cmMemPointLd : MemPointLd;
                    cmMemUnifLd  : MemUnifLd;
                    cmMemTripozLd : MemTripozLd;
                    cmChDir      : ChangeDir;
                    cmCascade    : Cascade7;
                    cmTile       : Tile7;
                    cmAbout      : About;
                    cmCalendar   : Calendar;
                    cmAsciiTab   : AsciiTab;
                    cmCalculator  : Calculator;
                    cmSaveDesktop : SaveDesktop;
                    cmRetrieveDesktop : RetrieveDesktop;
                    cmColors     : Colors;

```

```

        cmMouse          : Mouse;
        cmDosShell       : DosShell7;
        cmVideoMode      : VideoMode;
    else
        Exit;
    end;
    ClearEvent(Event);      end; end; end;
procedure TMyStructure.Idle;

var
    DataExists : pmyDataWindow;

function IsTileable(P: PView): Boolean; far;

begin
    IsTileable := P^.Options and ofTileable <> 0;
end;

begin
    TApplication.Idle;
    Clock^.Update;
    Heap^.Update;
    if Desktop^.FirstThat(@IsTileable) <> nil then
        EnableCommands([cmTile, cmCascade])
    else
        DisableCommands([cmTile, cmCascade]);
    DataExists := Message(Desktop, evBroadcast, cmDataExist, nil);
    if DataExists <> nil then
        EnableCommands (CommandSetData)
    else
        DisableCommands (CommandSetData);
end;

procedure TMyStructure.InitMenuBar;

var
    R: TRect;

begin
    GetExtent(R);
    R.B.Y := R.A.Y+1;
    MenuBar := New(PMenuBar, Init(R, NewMenu(
        NewSubMenu('~#15~', hcSystem, NewMenu(
            NewItem('~A-bout...', '', kbNoKey, cmAbout, hcSAbout,
            NewLine(
            NewItem('Ascii ~t~able', '', kbNoKey, cmAsciiTab, hcNoContext,
            NewItem('Ca~l~endar', '', kbNoKey, cmCalendar, hcNoContext,
            NewItem('~C~alculator', '', kbNoKey, cmCalculator, hcNoContext,
            NewItem('~V~ideo mode', '', kbNoKey, cmVideoMode, hcNoContext,
            nil)))))),
        NewSubMenu('~D~ata', hcData, NewMenu(
            NewItem('~N~ew', '', kbNoKey, cmDataNew, hcDNew,
            NewItem('~O~pen...', '', kbNoKey, cmDataOpen, hcDOpen,
            NewItem('~S~ave', '', kbNoKey, cmDataSave, hcDSave,
            NewItem('S~A~ve as...', '', kbNoKey, cmDataSaveAs, hcDSaveAs,
            NewItem('~P~rint File', '', kbNoKey, cmPrintText, hcNoContext,
            NewItem('~C~hange dir...', '', kbNoKey, cmChDir, hcFChangeDir,
            NewItem('~D~OS shell', '', kbNoKey, cmDosShell, hcFDosShell,
            NewItem('E~x~it', 'Alt-X', kbAltX, cmQuit, hcFExit, nil)))))),
        NewSubMenu('~E~dit', hcEdit, NewMenu(
            NewItem('~G~eneral Title', '', kbNoKey, cmDataGen, hcDGeneral,
            NewItem('~N~ode Geometry', '', kbNoKey, cmNodeGeom, hcNoContext,
            NewItem('Ma~t~erial Prop', '', kbNoKey, cmMatProperty, hcNoContext,
            NewItem('~C~rossSect Spec', '', kbNoKey, cmCrosSectSpec, hcNoContext,
            NewItem('~M~ember Data', '', kbNoKey, cmMemOrient, hcNoContext,
            NewItem('~R~estraints ', '', kbNoKey, cmRestraint, hcNoContext,
            nil)))))),
        NewSubMenu('~L~oad', hcNoContext, NewMenu(
            NewItem('~P~rescribed Displ. PDL', '', kbNoKey, cmMemPrsDisplc, hcNoContext,
            NewItem('~N~odal Load NPL', '', kbNoKey, cmNodPointLd, hcNoContext,
            NewItem('~M~ember Point Load MPL', '', kbNoKey, cmMemPointLd, hcNoContext,
            NewSubMenu('Member ~D~istributed Load', hcNoContext, NewMenu(
                NewItem('~U~niform Load MUL', '', kbNoKey, cmMemUnifLd, hcNoContext,
                NewItem('Tripo~z~ Load MTL', '', kbNoKey, cmMemTripozLd, hcNoContext, nil))),
            nil))))),
        NewSubMenu('Out~P~ut', hcOutput, NewMenu(
            NewItem('~O~pen...', 'F3', kbF3, cmFOpen, hcFOpen,

```

```

       NewItem('~S-ave As...', '', kbNoKey, cmSaveAs, hcFSaveAs,
        NewItem('~R-ename...', '', kbNoKey, cmRenamer, hcFRename,
        NewLine(
        NewItem('~C-hange dir...', '', kbNoKey, cmChDir, hcFChangeDir, nil))))),
NewSubMenu('~W-indows', hcWindows, NewMenu(
        NewItem('~R-esize/move', 'Ctrl-F5', kbCtrlF5, cmResize, hcWSizeMove,
        NewItem('~Z-oom', 'F5', kbF5, cmZoom, hcWZoom,
        NewItem('~N-ext', 'F6', kbF6, cmNext, hcWNext,
        NewItem('~P-revious', 'Shift-F6', kbShiftF6, cmPrev, hcWPrevious,
        NewItem('~T-ile', '', kbNoKey, cmTile, hcWTile,
        NewItem('~C-a-scade', '', kbNoKey, cmCascade, hcWCascade,
        NewItem('~C-lose', 'Alt-F3', kbAltF3, cmClose, hcWClose, nil)))))),
NewSubMenu('~O-ptions', hcOptions, NewMenu(
        NewItem('~M-ouse...', '', kbNoKey, cmMouse, hcOMouse,
        NewItem('~C-olors...', '', kbNoKey, cmColors, hcOColors,
        NewLine(
        NewItem('~S-ave desktop', '', kbNoKey, cmSaveDesktop, hcNoContext{0SaveDesktop}),
        NewItem('~R-etrieve desktop', '', kbNoKey, cmRetrieveDesktop,
        hcNoContext{0RetrieveDesktop}, nil))))), nil))))))));
end;

Procedure TMyStructure.InitStatusLine;

var
    R: TRect;

begin
    GetExtent(R);
    R.A.Y := R.B.Y - 1;
    StatusLine := New(PStatusLine, Init(R,
        NewStatusDef(0, $FFFF,
            NewStatusKey('~F1~ Help', kbF1, cmHelp,
            NewStatusKey('~Alt-F3~ Close', kbAltF3, cmClose,
            NewStatusKey('~F5~ Zoom', kbF5, cmZoom,
            NewStatusKey('~Alt-X~ Quit', kbAltX, cmquit,
            NewStatusKey('', kbF10, cmMenu,
            NewStatusKey('', kbCtrlF5, cmResize, nil))))), nil));
end;

Procedure TMyStructure.OutOfMemory;

begin
    MessageBox('Not enough memory available to complete operation.',
    nil, mfError + mfOkButton);
end;

Procedure TMyStructure.ViewData(FileName: PathStr);

var
    w:pmyDatawindow;

begin
    W := New(PmyDataWindow, Init(FileName));
    W^.HelpCtx := hcDataWindow;
    if ValidView(W) <> nil then
        begin
            Desktop^.Insert(W);
        end;
end;

Procedure TMyStructure.ViewFile(FileName: PathStr);

var
    w:pmyfilewindow;

begin
    W := New(PmyFileWindow, Init(FileName));
    W^.HelpCtx := hcOutput;
    if ValidView(W) <> nil then
        begin
            Desktop^.Insert(W);
            enableCommands([cmSaveAs]);
        end;
end;

Procedure TMyStructure.LoadDesktop(VAR S: TStream);
VAR
    P: PView;

```

```

Procedure CloseView(P: PView); far;
BEGIN
  Message(P, evCommand, cmClose, nil);
END;(*PROCEDURE*)

BEGIN
  IF Desktop^.Valid(cmClose) THEN
    BEGIN
      Desktop^.ForEach(@CloseView); { Clear the desktop }
      repeat
        P := PView(S.Get);
        Desktop^.InsertBefore(ValidView(P), Desktop^.Last);
      until P = nil;
    END;
  END;(*PROCEDURE TMyStructure.LoadDesktop*)

  Procedure TMyStructure.StoreDesktop(VAR S: TStream);

  Procedure WriteView(P: PView); far;
  BEGIN
    IF P <> Desktop^.Last THEN S.Put(P);
  END;(*PROCEDURE*)

  BEGIN
    Desktop^.ForEach(@WriteView);
    S.Put(nil);
  END;(*PROCEDURE TMyStructure.StoreDesktop*)

var
  Structure: TMyStructure;

begin
  ClrScr;
  Structure.Init;
  Structure.Run;
  Structure.Done;
end.

```

```

Program Analyse;
(*Warning: Before running this PROGRAM, reset the PC for memory allocation)
(*****)
(*          2D Frame Analysis          *)
(*          Version 1.0                 *)
(*          *)                          *)
(*          *)                          *)
(*          *)                          *)
(*  VERY IMPORTANT NOTES: 1) GLOBAL COORDINATES SHOULD BE SELECTED  *)
(*  SUCH THAT ALL THE STRUCTURE STAYS IN THE FIRST QUARTER OF PLANE. *)
(*  2) MEMBER ASSIGNMENT SHOULD BE SUCH THAT FI STAYS IN THIS RANGE. *)
(*  -0.5Pi<=FI<=+0.5Pi                                             *)
(*  This program analyses any type of 2D FRAMED STRUCTURE by elastic *)
(*  first-order stiffness matrix method, incorporating with different *)
(*  loadings, member connections, restraints, section module.      *)
(*  All footings are in a same level acting to the structure.      *)
(*  Assumption: Element transformation matrix is Orthogonal.      *)
(*  Restriction: Minimum of two nodes to be taken as footings.    *)
(*  Before running any example, user must make sure of structural  *)
(*  external stability.                                           *)
(*****)

($N+)
USES
  Crt, Dos, Build, Globals, SoilMode;
(*& data segment includes CONST, TYPE, VAR appeared in the build unit earlier*)

LABEL
  100;

PROCEDURE ShowMeReminderNote;
Begin
  Writeln('Before any attempt to run this program, ensure the following files');
  Writeln('exist in the working directory: Crt,Dos,Build,Globals,SprtData, ');
  Writeln('and all the Pascal and the binary forms of the structural input ');
  Writeln('data files listed in the Globals.pas also all the files associated');
  Writeln('with the Genforms.bat ...');
  Writeln('then ensure the format of directory''s path is compatible with the');
  Writeln('system is being used (i.e. c:\MyDucu~1\... or c:\My Documents\...');
  Writeln('press Enter to continue ...');
  Readln;
END;

(*& code segment starts here*)
PROCEDURE AllocateMemberDataConstants;
VAR
  j : LONGINT;  (*member counter*)
BEGIN
  FOR j := 1 TO NoMems DO BEGIN
    (*Elasticity unit conversion(*1e6) from GPa => KPa *)
    Emodul[j] :=Mat[Member[j].Mat].Elasticity_M * 1e6;
    Inr[j] :=Sec[Member[j].C_Sec].M_Inertia;
    L[j] :=Member[j].L;
    Area[j] :=Sec[Member[j].C_Sec].Area;
    Fi[j]:=Member[j].Fi;
  END;(* FOR j*)
END;(*PROCEDURE*)

PROCEDURE gaussj(VAR a: glnpbynp; n,np: LONGINT;
  VAR b: glnpbymp; m,mp: LONGINT);
VAR
  big,dum,pivinv: EXTENDED;
  i,icol,irow,j,k,l,ll: LONGINT;
  indxc,indxr,ipiv: glnp;
BEGIN
  FOR j := 1 to n DO BEGIN
    ipiv[j] := 0
  END;(*FORj*)
  FOR i := 1 to n DO BEGIN
    big := 0.0;
    FOR j := 1 to n DO BEGIN
      IF (ipiv[j] <> 1) THEN BEGIN
        FOR k := 1 to n DO BEGIN
          IF (ipiv[k] = 0) THEN BEGIN
            IF (abs(a[j,k]) >= big) THEN BEGIN
              big := abs(a[j,k]);
              irow := j;

```



```

        icol := k
    END
    END ELSE IF (ipiv[k] > 1) THEN BEGIN
        writeln('pause 1 in GAUSSJ - singular matrix');
        writeln(data1, 'pause 1 in GAUSSJ - singular
matrix'); Writeln(Chr(7)); EndProgram;
        readln
    END
    END
    END
END;
ipiv[icol] := ipiv[icol]+1;
IF (irow <> icol) THEN BEGIN
    FOR l := 1 to n DO BEGIN
        dum := a[irow,l];
        a[irow,l] := a[icol,l];
        a[icol,l] := dum
    END;
    FOR l := 1 to m DO BEGIN
        dum := b[irow,l];
        b[irow,l] := b[icol,l];
        b[icol,l] := dum
    END
END;
indxr[i] := irow;
indxc[i] := icol;
IF (a[icol,icol] = 0.0) THEN BEGIN
    writeln('pause 2 in GAUSSJ - singular matrix');
    writeln(data1, 'pause 2 in GAUSSJ - singular matrix');
    Writeln(Chr(7)); EndProgram; readln
END;
pivinv := 1.0/a[icol,icol];
a[icol,icol] := 1.0;
FOR l := 1 to n DO BEGIN
    a[icol,l] := a[icol,l]*pivinv
END;
FOR l := 1 to m DO BEGIN
    b[icol,l] := b[icol,l]*pivinv
END;
FOR ll := 1 to n DO BEGIN
    IF (ll <> icol) THEN BEGIN
        dum := a[ll,icol];
        a[ll,icol] := 0.0;
        FOR l := 1 to n DO BEGIN
            a[ll,l] := a[ll,l]-a[icol,l]*dum
        END;
        FOR l := 1 to m DO BEGIN
            b[ll,l] := b[ll,l]-b[icol,l]*dum
        END
    END
END
END;
FOR l := n DOWNT0 1 DO BEGIN
    IF (indxr[l] <> indxc[l]) THEN BEGIN
        FOR k := 1 to n DO BEGIN
            dum := a[k,indxr[l]];
            a[k,indxr[l]] := a[k,indxc[l]];
            a[k,indxc[l]] := dum
        END
    END
END
END
END; (*PROCEDUR*)

PROCEDURE DetermineElementTransMatrices;
VAR
    j, k, m : LONGINT;
BEGIN
    FOR j := 1 TO NoMems DO BEGIN
        (* transformation matrix: row 1 *)
        ElemTransMat[j,1,1] := Cos(Fi[j]);
        if ABS(ElemTransMat[j,1,1])=0 then ElemTransMat[j,1,1]:=0;
        ElemTransMat[j,1,2] := Sin(Fi[j]);
        if ABS(ElemTransMat[j,1,2])=0 then ElemTransMat[j,1,2]:=0;
        ElemTransMat[j,1,3] := 0;
        ElemTransMat[j,1,4] := 0;
        ElemTransMat[j,1,5] := 0;
        ElemTransMat[j,1,6] := 0;
    END
END

```

```

(* transformation matrix: row 2 *)
ElemTransMat[j,2,1] := -Sin(Fi[j]);
  if ABS(ElemTransMat[j,2,1])=0 then ElemTransMat[j,2,1]:=0;
ElemTransMat[j,2,2] := Cos(Fi[j]);
  if ABS(ElemTransMat[j,2,2])=0 then ElemTransMat[j,2,2]:=0;
ElemTransMat[j,2,3] := 0;
ElemTransMat[j,2,4] := 0;
ElemTransMat[j,2,5] := 0;
ElemTransMat[j,2,6] := 0;

(* transformation matrix: row 3 *)
ElemTransMat[j,3,1] := 0;
ElemTransMat[j,3,2] := 0;
ElemTransMat[j,3,3] := 1;
ElemTransMat[j,3,4] := 0;
ElemTransMat[j,3,5] := 0;
ElemTransMat[j,3,6] := 0;

(* transformation matrix: row 4 *)
ElemTransMat[j,4,1] := 0;
ElemTransMat[j,4,2] := 0;
ElemTransMat[j,4,3] := 0;
ElemTransMat[j,4,4] := Cos(Fi[j]);
  if ABS(ElemTransMat[j,4,4])=0 then ElemTransMat[j,4,4]:=0;
ElemTransMat[j,4,5] := Sin(Fi[j]);
  if ABS(ElemTransMat[j,4,5])=0 then ElemTransMat[j,4,5]:=0;
ElemTransMat[j,4,6] := 0;

(* transformation matrix: row 5 *)
ElemTransMat[j,5,1] := 0;
ElemTransMat[j,5,2] := 0;
ElemTransMat[j,5,3] := 0;
ElemTransMat[j,5,4] := -Sin(Fi[j]);
  if ABS(ElemTransMat[j,5,4])=0 then ElemTransMat[j,5,4]:=0;
ElemTransMat[j,5,5] := Cos(Fi[j]);
  if ABS(ElemTransMat[j,5,5])=0 then ElemTransMat[j,5,5]:=0;
ElemTransMat[j,5,6] := 0;

(* transformation matrix: row 6 *)
ElemTransMat[j,6,1] := 0;
ElemTransMat[j,6,2] := 0;
ElemTransMat[j,6,3] := 0;
ElemTransMat[j,6,4] := 0;
ElemTransMat[j,6,5] := 0;
ElemTransMat[j,6,6] := 1;
END;(*FOR j*)
FOR j := 1 TO NoMems DO BEGIN
  writeln(data1);
  writeln(data1,'Transformation matrix for member number :', j:3);
  writeln(data1);
  FOR k := 1 TO 6 DO BEGIN
    FOR m := 1 TO 6 DO
      write(data1,ElemTransMat[j,k,m]:8:3,char(9));
    writeln(data1);
  END;(*FOR k*)
END;(*FOR j*)
END;(*PROCEDURE*)

PROCEDURE DetermineLocalElementStiffnessMatrices;
(*to determine the local element stiffness matrix*)
VAR
  j, k, m, FACT : LONGINT;
  EA_L, EI_L3 : EXTENDED;
BEGIN
  EA_L:=0; EI_L3:=0;
  FOR j := 1 TO NoMems DO BEGIN
    Fact :=0;
    S_J := Member[j].S_Join;
    E_J := Member[j].E_Join;
    EA_L := Emodul[j]*Area[j]/L[j]; (*KPa*)
    EI_L3 := Emodul[j]*Inr[j]/(L[j]*L[j]*L[j]); (*KPa*)

    (* stiffness matrix: row 1 *)
    ElemL_StiffMat[j,1,1] := EA_L;
    ElemL_StiffMat[j,1,2] := 0;
    ElemL_StiffMat[j,1,3] := 0;

```

```

ElemL_StiffMat[j,1,4] := -EA_L;
ElemL_StiffMat[j,1,5] := 0;
ElemL_StiffMat[j,1,6] := 0;

(* stiffness matrix: row 2 *)
ElemL_StiffMat[j,2,1] := 0;
If (S_J=0)and (E_J=1) then FACT := 3 else
If (S_J=1)and (E_J=0) then FACT := 3 else
If (S_J=0)and (E_J=0) then FACT := 12;
ElemL_StiffMat[j,2,2] := FACT*EI_L3;
If (S_J=0)and (E_J=1) then FACT := 3 else
If (S_J=1)and (E_J=0) then FACT := 0 else
If (S_J=0)and (E_J=0) then FACT := 6;
ElemL_StiffMat[j,2,3] := FACT*EI_L3*L[j];
ElemL_StiffMat[j,2,4] := 0;
If (S_J=0)and (E_J=1) then FACT := -3 else
If (S_J=1)and (E_J=0) then FACT := -3 else
If (S_J=0)and (E_J=0) then FACT := -12;
ElemL_StiffMat[j,2,5] := FACT*EI_L3;
If (S_J=0)and (E_J=1) then FACT := 0 else
If (S_J=1)and (E_J=0) then FACT := 3 else
If (S_J=0)and (E_J=0) then FACT := 6;
ElemL_StiffMat[j,2,6] := FACT*EI_L3*L[j];

(* stiffness matrix: row 3 *)
ElemL_StiffMat[j,3,1] := 0;
If (S_J=0)and (E_J=1) then FACT := 3 else
If (S_J=1)and (E_J=0) then FACT := 0 else
If (S_J=0)and (E_J=0) then FACT := 6;
ElemL_StiffMat[j,3,2] := FACT*EI_L3*L[j];
If (S_J=0)and (E_J=1) then FACT := 3 else
If (S_J=1)and (E_J=0) then FACT := 0 else
If (S_J=0)and (E_J=0) then FACT := 4;
ElemL_StiffMat[j,3,3] := FACT*EI_L3*L[j]*L[j];
ElemL_StiffMat[j,3,4] := 0;
If (S_J=0)and (E_J=1) then FACT := -3 else
If (S_J=1)and (E_J=0) then FACT := 0 else
If (S_J=0)and (E_J=0) then FACT := -6;
ElemL_StiffMat[j,3,5] := FACT*EI_L3*L[j];
If (S_J=0)and (E_J=1) then FACT := 0 else
If (S_J=1)and (E_J=0) then FACT := 0 else
If (S_J=0)and (E_J=0) then FACT := 2;
ElemL_StiffMat[j,3,6] := FACT*EI_L3*L[j]*L[j];

(* stiffness matrix: row 4 *)
ElemL_StiffMat[j,4,1] := -EA_L;
ElemL_StiffMat[j,4,2] := 0;
ElemL_StiffMat[j,4,3] := 0;
ElemL_StiffMat[j,4,4] := EA_L;
ElemL_StiffMat[j,4,5] := 0;
ElemL_StiffMat[j,4,6] := 0;

(* stiffness matrix: row 5 *)
ElemL_StiffMat[j,5,1] := 0;
If (S_J=0)and (E_J=1) then FACT := -3 else
If (S_J=1)and (E_J=0) then FACT := -3 else
If (S_J=0)and (E_J=0) then FACT := -12;
ElemL_StiffMat[j,5,2] := FACT*EI_L3;
If (S_J=0)and (E_J=1) then FACT := -3 else
If (S_J=1)and (E_J=0) then FACT := 0 else
If (S_J=0)and (E_J=0) then FACT := -6;
ElemL_StiffMat[j,5,3] := FACT*EI_L3*L[j];
ElemL_StiffMat[j,5,4] := 0;
If (S_J=0)and (E_J=1) then FACT := 3 else
If (S_J=1)and (E_J=0) then FACT := 3 else
If (S_J=0)and (E_J=0) then FACT := 12;
ElemL_StiffMat[j,5,5] := FACT*EI_L3;
If (S_J=0)and (E_J=1) then FACT := 0 else
If (S_J=1)and (E_J=0) then FACT := -3 else
If (S_J=0)and (E_J=0) then FACT := -6;
ElemL_StiffMat[j,5,6] := FACT*EI_L3*L[j];

(* stiffness matrix: row 6 *)
ElemL_StiffMat[j,6,1] := 0;
If (S_J=0)and (E_J=1) then FACT := 0 else
If (S_J=1)and (E_J=0) then FACT := 3 else
If (S_J=0)and (E_J=0) then FACT := 6;

```

```

ElemL_StiffMat[j,6,2] := FACT*EI_L3*L[j];
If (S_J=0)and (E_J=1) then FACT := 0 else
If (S_J=1)and (E_J=0) then FACT := 0 else
If (S_J=0)and (E_J=0) then FACT := 2;
ElemL_StiffMat[j,6,3] := FACT*EI_L3*L[j]*L[j];
ElemL_StiffMat[j,6,4] := 0;
If (S_J=0)and (E_J=1) then FACT := 0 else
If (S_J=1)and (E_J=0) then FACT := -3 else
If (S_J=0)and (E_J=0) then FACT := -6;
ElemL_StiffMat[j,6,5] := FACT*EI_L3*L[j];
If (S_J=0)and (E_J=1) then FACT := 0 else
If (S_J=1)and (E_J=0) then FACT := 3 else
If (S_J=0)and (E_J=0) then FACT := 4;
ElemL_StiffMat[j,6,6] := FACT*EI_L3*L[j]*L[j];
END; (*FOR j*)

Writeln(datal);
writeln(datal);
FOR j := 1 TO NoMems DO BEGIN
  writeln(datal);
  writeln(datal,'Element Local Stiffness matrix for member number:',
    j:3);
  writeln(datal);
  FOR k := 1 TO 6 DO Begin
    FOR m := 1 TO 6 DO BEGIN (*KPa*)
      write(datal,ElemL_StiffMat[j,k,m]:8:3,char(9));
    END; (*FOR k*)
  writeln(datal);
  End; (*for k16*)
  END; (*FOR j*)
  writeln(datal);
END; (*PROCEDURE*)

PROCEDURE DetermineElemG_StiffMat;
VAR
  j, (* member counter*)
  k, (* row counter premultiplier *)
  m, (* row counter postmultiplier *)
  n : LONGINT; (* col counter premultiplier & row counter postmultiplier *)
BEGIN
  (* rem:Temp[k,m] = ElemL_StiffMat[k,n] x ElemTransMat[n,m] for j members. *)
  FOR j :=1 to NoMems DO BEGIN
    FOR k :=1 TO 6 DO (*row counter premultiplier *)
      FOR m :=1 TO 6 DO
        FOR n :=1 TO 6 DO begin
          Temp[j,k,m] := ElemL_StiffMat[j,k,n]*ElemTransMat[j,n,m] + Temp[j,k,m];
        end; (*for n *)
      END; (*for j *)
    END; (*for k *)
  END; (*for j *)

  writeln(datal);
  writeln(datal);
  FOR j := 1 TO NoMems DO BEGIN
    writeln(datal);
    writeln(datal,'Element TEMP matrix for member number :', j:3);
    writeln(datal);
    FOR k := 1 TO 6 DO Begin
      FOR m := 1 TO 6 DO BEGIN
        write(datal,TEMP[j,k,m]:8:3,char(9));
      END; (*FOR m*)
    END; (*FOR k*)
    writeln(datal);
    End; (*for k16*)
  END; (*FOR j*)
  writeln(datal);

  (* Transpose ElemTransMat[n,m] for j members due to being orthogonal *)
  FOR j :=1 TO NoMems DO BEGIN
    FOR k := 1 TO 6 DO
      FOR m :=1 TO 6 DO
        IF k<>m THEN begin
          ElemTposTransMat[j,k,m] := - ElemTransMat[j,k,m];
          if ABS(ElemTposTransMat[j,k,m]) = 0 then
            ElemTposTransMat[j,k,m] := 0;
          end ELSE
            ElemTposTransMat[j,k,m] := ElemTransMat[j,k,m];
        END; (* for j1NoMems*)
      END;
    END;
  END;
  writeln(datal);

```

```

writeln(data1);
FOR j := 1 TO NoMems DO BEGIN
  writeln(data1);
  writeln(data1, 'Element Transpose Transformation matrix for ',
    'member number :', j:3);
  writeln(data1);
  FOR k := 1 TO 6 DO BEGIN
    FOR m := 1 TO 6 DO BEGIN
      write(data1, ElemTposTransMat[j,k,m]:8:3, char(9));
    end; (*FOR m*)
    writeln(data1);
  END; (*FOR k*)
END; (*FOR j*)
writeln(data1);

(*rem: ElemG_StiffMat[j,k,m]=ElemTposTransMat[k,m] x Temp[k,m] for j elements.*)
FOR j :=1 to NoMems DO BEGIN
  FOR k :=1 TO 6 DO (*row counter premultiplier*)
    FOR m :=1 TO 6 DO
      FOR n :=1 TO 6 DO begin (*KPa*)
        ElemG_StiffMat[j,k,m] :=
          ElemG_StiffMat[j,k,m]+ElemTposTransMat[j,k,n] * Temp[j,n,m];
      end; (* for k & m & n*)
    END; (* for j*)

  writeln(data1);
  writeln(data1);
  FOR j := 1 TO NoMems DO BEGIN
    writeln(data1);
    writeln(data1, 'Element Global Stiff matrix for member number :',
      j:3);
    writeln(data1);
    FOR k := 1 TO 6 DO BEGIN (*KPa*)
      FOR m := 1 TO 6 DO BEGIN
        write(data1, ElemG_StiffMat[j,k,m]:8:3, char(9));
      END; (*FOR m*)
    END; (* for k16*)
    writeln(data1);
    END; (*FOR j*)
  writeln(data1);
END; (* procedure*)

PROCEDURE DetermineGlobalStiffnessMatrix;
VAR
  j,
  k, m, (*element matrix counters*)
  S, (*start node at GSM*)
  E : LONGINT; (*end node at GSM*)
BEGIN
  S:=0; E:=0; (*initialising temporary parameters*)

  (* arranging Global Stiffness Matrix for non-consecutive nodes setting*)
  BEGIN
    FOR j := 1 TO NoMems DO BEGIN
      S:= Member[j].S_Node;
      E:= Member[j].E_Node;

      FOR k := 1 TO 3 DO BEGIN
        FOR m := 1 TO 3 DO BEGIN
          GlobalStifMat^[3*(S-1)+k,3*(S-1)+m]:=
            GlobalStifMat^[3*(S-1)+k,3*(S-1)+m] + ElemG_StiffMat[j,k,m];
        END; (*FOR m13 *)
        FOR m := 4 TO 6 DO BEGIN
          GlobalStifMat^[3*(S-1)+k,3*(E-1)+(m-3)]:=
            GlobalStifMat^[3*(S-1)+k,3*(E-1)+(m-3)]+ElemG_StiffMat[j,k,m];
        END; (*FOR m46 *)
      END; (*FOR k13 *)

      FOR k := 4 TO 6 DO BEGIN
        FOR m := 1 TO 3 DO BEGIN
          GlobalStifMat^[3*(E-1)+(k-3),3*(S-1)+(m)]:=
            GlobalStifMat^[3*(E-1)+(k-3),3*(S-1)+(m)] +
            ElemG_StiffMat[j,k,m];
        END; (*FOR m13 *)
        FOR m := 4 TO 6 DO BEGIN
          GlobalStifMat^[3*(E-1)+(k-3),3*(E-1)+(m-3)]:=
            GlobalStifMat^[3*(E-1)+(k-3),3*(E-1)+(m-3)] +

```

```

        ElemG_StiffMat[j,k,m];
    END;    (*FOR m46 *)
    END;    (*FOR k46 *)
    END;    (* FOR j*)
    END;
    Writeln(data1);
    Writeln(data1);
    Writeln(data1,' Structure Global Stiffness Matrix with NO-SPRING RESTRAINS ');
    Writeln(data1);
    FOR k:= 1 TO 3*NoNodes DO BEGIN
        FOR m := 1 TO 3*NoNodes DO begin            (*KPa*)
            Write(data1,GlobalStifMat^[k,m]:8:3,char(9));
        end;    (*FOR m13*NN*)
        Writeln(data1);
        Writeln(data1);
    END;    (*FOR k13*NN *)

    Writeln(data1);

    k := 0;
    FOR j := 1 TO NoNdRs DO BEGIN
        k:= NdRs[j].name;                (*Restrained Node is identified*)

        IF (NdRs[j].Xdir = 2) THEN BEGIN
            GlobalStifMat^[3*k-2,3*k-2] (*KPa*) :=
            NdRs[j].SprX(*KPa*) + GlobalStifMat^[3*k-2,3*k-2]; (*KPa*)
            writeln(data1,'X Restrain at node:',k:3,' column:',(3*k-2):3,
            char(32),NdRs[j].SprX:20:5);
        END;    (*IF X*)

        IF (NdRs[j].Ydir = 2) THEN BEGIN
            GlobalStifMat^[3*k-1,3*k-1] :=
            NdRs[j].SprY + GlobalStifMat^[3*k-1,3*k-1];
            writeln(data1,'Y Restrain at node:',k:3,' column:',(3*k-1):3,
            char(32),NdRs[j].SprY:20:5);
        END;    (*IF Y*)

        IF (NdRs[j].Zrot = 2) THEN BEGIN
            GlobalStifMat^[3*k,3*k] :=
            NdRs[j].SprZ + GlobalStifMat^[3*k,3*k];
            writeln(data1,'Z Restrain at node:',k:3,' column:',(3*k):3,
            char(32),NdRs[j].SprZ:20:5);
        END;    (*IF Z*)

    END;    (*FOR j1NN*)

    Writeln(data1);
    Writeln(data1);
    Writeln(data1,'Global SuperStructure Stiffness Matrix SPRING RESTRAINS INCLUDED ');
    Writeln(data1);
    FOR k:= 1 TO 3*NoNodes DO BEGIN
        FOR m := 1 TO 3*NoNodes DO begin            (*KPa*)
            Write(data1,GlobalStifMat^[k,m]:12:6,char(9));
        end;    (*FOR m13*NN*)
        Writeln(data1);
        Writeln(data1);
    END;    (*FOR k13*NN*)
    Writeln(data1);
END;    (*PROCEDURE*)

PROCEDURE DetermineSoilStiffnessMatrix;
PROCEDURE InvertMatrix1(NSFlexMat:TwoDMatrixF;var SStifMat: TwoDMatrixF;
SizOfSqrMat:LongInt);
(*Program Inverse SoilFlexMat^(originally from Fortran version inversAmat.txt;*)
(*This procedure is calculating the soil stiffness matrix from inversion *)
(*of the soil flexibility matrix then named as soilstifmatrix*)

Label
    10;
Var
    i, j, J9, k, K9, N, N9, p, q: Longint;
    D, DI: EXTENDED;
begin
    i:=0; j:=0; J9:=0; k:=0; k9:=0; N:=0; N9:=0; D:=0; DI:=0;
    N := EHS_dor;    (*Size of EHS degrees of Restrain(non-zero) square matrix*)
    N9 := N-1;
    For i := 1 TO N Do begin

```

```

DI := NSFlexMat[i,1];
If ( DI=0 ) Then begin
  Writeln(data1,'Singular Matrix in soil flexi inversion');
  Writeln(Chr(7));Readln; EndProgram; Halt; End;  (*if (DI=0)*)
For j:=1 TO N9 Do begin
  J9 := j+1;
  NSFlexMat[i,j]:= NSFlexMat[i,J9] / DI;
end;      (*j1!n9*)
NSFlexMat[i,N]:= 1 / DI;
For j:= 1 TO N Do begin
  If (j =i) Then GoTo 10;
  D := NSFlexMat[j,1];
  For k :=1 TO N9 Do begin
    k9 := k +1;
    NSFlexMat[j,k]:= NSFlexMat[j,k9] - NSFlexMat[i,k] * D;
  end;      (*fork!1N9*)
  NSFlexMat[j,N]:= -NSFlexMat[i,N] * D;
10:   end;      (*forj!1N*)
end;      (*fori1N*)
(*result of inversion, has been stored in the same variable "SoilFlexMat
and then copied in "SoilStifMat"*)
SStifMat:= NSFlexMat;
END; (*procedure inve...1*)

PROCEDURE PrintSoilStifMat1(var data1:TEXT);
var
  p, q :Longint;
begin
  Writeln(data1,' Soil Stiffness Matrix (all blocks covered) :');
  Writeln(data1);
  FOR p := 1 TO EHS_dor DO BEGIN
    FOR q := 1 TO EHS_dor DO BEGIN
      Write(data1,SoilStifMat^[p,q]:10:6,chr(9));
    END; (*FOR i!1n*)
    Writeln(data1);
    Writeln(data1);
  End; (*FOR j!1n*)
END;      (*procedure prin...1*)

PROCEDURE InvertMatrix2 (x:TwoDMatrixF; var InvX:TwoDMatrixF; dor:LongInt);
var
  i, j : LongInt;
begin
  (*inversion of a diagonal matrix*)
  for i:= 1 to dor do
    for j:= 1 to dor do
      if (i=j) then begin
        InvX[i,j]:= 1/x[i,j];
      end;
    end;
  END;

PROCEDURE PrintSoilStifMat2(var data1:TEXT);
var
  p, q :Longint;
begin
  Writeln(data1,' Soil Stiffness Matrix (Diagonal blocks are considered):');
  Writeln(data1);
  FOR p := 1 TO EHS_dor DO BEGIN
    FOR q := 1 TO EHS_dor DO BEGIN
      Write(data1,SoilStifMat^[p,q]:10:6,chr(9));
    END; (*FOR i!1n*)
    Writeln(data1);
    Writeln(data1);
  End; (*FOR j!1n*)
end;      (*procedure prin...2*)

BEGIN
  IF (ConsiderAlone in NO) then begin
    InvertMatrix1(NetSoilFlexMat^, SoilStifMat^,EHS_dor);
    PrintSoilStifMat1(data1);
  End
  ELSE BEGIN
    InvertMatrix2.(NetSoilFlexMat^, SoilStifMat^,EHS_dor);
    PrintSoilStifMat2(data1);
  End;
END;      (*procedure*)

PROCEDURE DetermineCombinedMatrices;

```

```

VAR
  i, j, m, n : Longint;
  (* this global var: "CombSoilStrucStifMat" is onward used for calculation *)
BEGIN
  For i := 1 TO 3*NoNodes DO
    For j := 1 TO 3*NoNodes DO
      CombinSoilStrucStifMat^[i,j] := GlobalStifMat^[i,j];

  If SelectedSoilModel <> 0 then
    For i := 1 TO EHS_dor DO Begin
      m := SoilPosDOR[i].GlobStif;
      For j := 1 TO EHS_dor DO Begin
        n := SoilPosDOR[j].GlobStif;
        CombinSoilStrucStifMat^[m,n] :=
          SoilStifMat^[i,j] + CombinSoilStrucStifMat^[m,n];
      End; (* for j!1EHS_dor *)
    End; (* for i!1EHS_dor *)

    Writeln(data1, ' Combined Soil Structure Stiffness Matrix : ');
    Writeln(data1);
    FOR i := 1 TO 3*NoNodes DO BEGIN
      FOR j := 1 TO 3*NoNodes DO BEGIN
        Write(data1, CombinSoilStrucStifMat^[i,j]:12:6, chr(9));
      END; (*FOR i!1n *)
    Writeln(data1);
    Writeln(data1);
  END; (*FOR j!1n *)
END; (*procedure*)

PROCEDURE DetermineLocalFixedEndForces;
VAR
  j, k, m : LONGINT;
BEGIN
  (*transfer the loads Global => Local, then
  do the calculation for the fixed end forces,
  finally transfer the FEFvector into the global system.*)

  (* Member Point Loads *)
  (* Note: ONLY FORCE & MOMENT values are in Global sys
  (* (ie: X & Y ); but XLoc is in local sys(ie: x' & y')
  (* [ Cos fi Sin fi ] { FX } {Fx'
  (* { | * | = | |
  (* [-Sin fi Cos fi ] { FY } {Fy'
  (* ( calculate the FEF's in the local system *)

  IF (NoPtLd > 0) AND (PtLd[1].mem > 0) THEN BEGIN
    For j:= 1 TO NoPtLd DO BEGIN
      m := PtLd[j].mem;
      PtLd[j].Fx_L := ElemTransMat[m,1,1]*PtLd[j].Fx +
        ElemTransMat[m,1,2]*PtLd[j].Fy + PtLd[j].Fx_L;
      PtLd[j].Fy_L := ElemTransMat[m,2,1]*PtLd[j].Fx +
        ElemTransMat[m,2,2]*PtLd[j].Fy + PtLd[j].Fy_L;
    END;

    For j:= 1 TO NoPtLd DO BEGIN
      m := PtLd[j].mem;
      S_J := Member[m].S_Join;
      E_J := Member[m].E_Join;

      (* axial force at left hand end *)
      ElemFxdFce_L[m,1] := -PtLd[j].Fx_L*(1-PtLd[j].XLoc/L[m]) +
        ElemFxdFce_L[m,1];

      (* shear force at left hand end *)
      If (S_J=0) and (E_J=1) then
        ElemFxdFce_L[m,2] := (-PtLd[j].Fy_L/2)*(1-PtLd[j].XLoc/L[m])*
          (3-sqr(1-PtLd[j].XLoc/L[m])) +
          (3*PtLd[j].Mz/2/L[m])*(PtLd[j].XLoc/L[m])*
          (2-PtLd[j].XLoc/L[m]) +
          ElemFxdFce_L[m,2]
      else
        If (S_J=1) and (E_J=0) then
          ElemFxdFce_L[m,2] := (-PtLd[j].Fy_L/2)*sqr(1-PtLd[j].XLoc/L[m])*
            (2+PtLd[j].XLoc/L[m]) +
            (3*PtLd[j].Mz/2/L[m])*(1-PtLd[j].XLoc/L[m])*
            (1+PtLd[j].XLoc/L[m]) +

```



```

ElemFxdFce_L[m,2]
else
If (S_J=0)and (E_J=0) then
ElemFxdFce_L[m,2] := -PtLd[j].Fy_L*sqr(1-PtLd[j].XLoc/L[m])*
(3-2*(1-PtLd[j].XLoc/L[m])) +
6*PtLd[j].Mz*(PtLd[j].XLoc/L[m])*
(1-PtLd[j].XLoc/L[m])/L[m] +
ElemFxdFce_L[m,2]
else
If (S_J=1)and (E_J=1) then
ElemFxdFce_L[m,2] := -PtLd[j].Fy_L*(1-PtLd[j].XLoc/L[m]) +
PtLd[j].Mz/L[m] +
ElemFxdFce_L[m,2];

(* moment at left hand end *)
If (S_J=0)and (E_J=1) then
ElemFxdFce_L[m,3] := -PtLd[j].Fy_L*L[m]*(PtLd[j].XLoc/L[m])*
(1-PtLd[j].XLoc/L[m])*(2-PtLd[j].XLoc/L[m])/2 +
PtLd[j].Mz*(1-3*sqr(1-PtLd[j].XLoc/L[m]))/2 +
ElemFxdFce_L[m,3]
else
If (S_J=1)and (E_J=0) then
ElemFxdFce_L[m,3] := ElemFxdFce_L[m,3]
else
If (S_J=0)and (E_J=0) then
ElemFxdFce_L[m,3] := -PtLd[j].Fy_L*L[m]*(PtLd[j].XLoc/L[m])*
sqr(1-PtLd[j].XLoc/L[m]) +
PtLd[j].Mz*(1-PtLd[j].XLoc/L[m])*
(2-3*(1-PtLd[j].XLoc/L[m])) +
ElemFxdFce_L[m,3]
else
If (S_J=1)and (E_J=1) then
ElemFxdFce_L[m,3] := ElemFxdFce_L[m,3];

(* axial force at right hand end *)
ElemFxdFce_L[m,4] := -PtLd[j].Fx_L*(PtLd[j].XLoc/L[m]) +
ElemFxdFce_L[m,4];

(* shear force at right hand end *)
If (S_J=0)and (E_J=1) then
ElemFxdFce_L[m,5] := -PtLd[j].Fy_L*sqr(PtLd[j].XLoc/L[m])*
(3-PtLd[j].XLoc/L[m])/2 -
3*PtLd[j].Mz*(PtLd[j].XLoc/L[m])*
(2-PtLd[j].XLoc/L[m])/2/L[m] +
ElemFxdFce_L[m,5]
else
If (S_J=1)and (E_J=0) then
ElemFxdFce_L[m,5] := -PtLd[j].Fy_L*(PtLd[j].XLoc/L[m])*
(3-sqr(PtLd[j].XLoc/L[m]))/2 +
(-3*PtLd[j].Mz)*(1-PtLd[j].XLoc/L[m])*
(1+PtLd[j].XLoc/L[m])/2/L[m] +
ElemFxdFce_L[m,5]
else
If (S_J=0)and (E_J=0) then
ElemFxdFce_L[m,5] := -PtLd[j].Fy_L*sqr(PtLd[j].XLoc/L[m])*
(3-2*PtLd[j].XLoc/L[m]) -
6*PtLd[j].Mz*PtLd[j].XLoc/L[m]*
(1-PtLd[j].XLoc/L[m])/L[m] +
ElemFxdFce_L[m,5]
else
If (S_J=1)and (E_J=1) then
ElemFxdFce_L[m,5] := -PtLd[j].Fy_L*(PtLd[j].XLoc/L[m]) +
PtLd[j].Mz/L[m] + ElemFxdFce_L[m,5];

(* moment at right hand end *)
If (S_J=0)and (E_J=1) then
ElemFxdFce_L[m,6] := ElemFxdFce_L[m,6]
else
If (S_J=1)and (E_J=0) then
ElemFxdFce_L[m,6] := PtLd[j].Fy_L*L[m]*(PtLd[j].XLoc/L[m])*
(1-PtLd[j].XLoc/L[m])*(1+PtLd[j].XLoc/L[m])/2 +
PtLd[j].Mz*(1-3*sqr(PtLd[j].XLoc/L[m]))/2 +
ElemFxdFce_L[m,6]
else
If (S_J=0)and (E_J=0) then
begin
ElemFxdFce_L[m,6] := PtLd[j].Fy_L*L[m]*(1-PtLd[j].XLoc/L[m])*

```

```

                                sqr(PtLd[j].XLoc/L[m]) +
                                PtLd[j].Mz*(PtLd[j].XLoc/L[m])*
                                (2-3*(PtLd[j].XLoc/L[m])) +
                                ElemFxdFce_L[m,6]

                                end

                                else
                                If (S_J=1)and (E_J=1) then
                                    ElemFxdFce_L[m,6]:= ElemFxdFce_L[m,6];
                                END;      (*FOR*)

                                Writeln(data1);
                                Writeln(data1);
                                Writeln(data1,'Element Fixed End Forces in Local System by Point Loads only :');
                                Writeln(data1,'F"x@s':11,char(32),'F"y@s':11,char(32),'Mz@s':11,
                                    char(32),
                                    'F"x@e':11,char(32),'F"y@e':11,char(32),'Mz@e':11);
                                Writeln(data1);
                                Writeln(data1,' Fixed end forces for members:');
                                FOR m := 1 TO NoMems DO BEGIN
                                    Writeln(data1,'Member:',m:3);
                                    FOR k:= 1 TO 6 DO BEGIN
                                        Write(data1,ElemFxdFce_L[m,k]:11:4,char(32));
                                    END;      (*FOR k16*)
                                    Writeln(data1);
                                END;      (*FOR j1NoMem*)
                                Writeln(data1);
                                END;      (* IF NoPtLd>0*)

                                (* Uniformly Distributed Loads *)
                                IF (NoUDistLd >0) AND (UDistLd[1].mem >0) THEN BEGIN
                                    m :=0;S_J:=0;E_J:=0;
                                    For j:= 1 TO NoUDistLd DO BEGIN
                                        m := UDistLd[j].mem;
                                        if Fi[m]>0 then begin      (*for Fi in the first quarter*)
                                            UDistLd[j].Ux_L := Sin(Fi[m])*Cos(Fi[m])*
                                                (UDistLd[j].Ux+UDistLd[j].Uy) +
                                                UDistLd[j].Ux_L;
                                            UDistLd[j].Uy_L := -sqr(Sin(Fi[m]))*UDistLd[j].Ux +
                                                sqr(Cos(Fi[m]))*UDistLd[j].Uy +
                                                UDistLd[j].Uy_L;
                                        end else
                                        if Fi[m]<0 then begin      (*for Fi in the fourth quarter*)
                                            UDistLd[j].Ux_L := Sin(Fi[m])*Cos(Fi[m])*
                                                (-UDistLd[j].Ux+UDistLd[j].Uy) +
                                                UDistLd[j].Ux_L;
                                            UDistLd[j].Uy_L := sqr(Sin(Fi[m]))*UDistLd[j].Ux +
                                                sqr(Cos(Fi[m]))*UDistLd[j].Uy +
                                                UDistLd[j].Uy_L;
                                        end else
                                        begin      (*for Fi=0 a horizontal member*)
                                            UDistLd[j].Ux_L := ElemTransMat[m,1,1]*UDistLd[j].Ux +
                                                ElemTransMat[m,1,2]*UDistLd[j].Uy +
                                                UDistLd[j].Ux_L;
                                            UDistLd[j].Uy_L := ElemTransMat[m,2,1]*UDistLd[j].Ux +
                                                ElemTransMat[m,2,2]*UDistLd[j].Uy +
                                                UDistLd[j].Uy_L;
                                        end;
                                    END;

                                    FOR j := 1 TO NoUDistLd DO BEGIN
                                        m := UDistLd[j].mem;
                                        S_J := Member[m].S_Join;
                                        E_J := Member[m].E_Join;

                                        (* axial force at left hand end *)
                                        U_ElemFxdFce_L[m,1]:= -UDistLd[j].Ux_L*L[m]/2 +
                                            U_ElemFxdFce_L[m,1];

                                        (* shear force at left hand end *)
                                        If (S_J=0)and (E_J=1) then
                                            U_ElemFxdFce_L[m,2]:= -5*UDistLd[j].Uy_L*L[m]/8 +
                                                U_ElemFxdFce_L[m,2]
                                        else
                                            If (S_J=1)and (E_J=0) then
                                                U_ElemFxdFce_L[m,2]:= -3*UDistLd[j].Uy_L*L[m]/8 +

```

```

                                U_ElemFxdFce_L[m,2]
else
If (S_J=0)and (E_J=0) OR ((S_J=1)and (E_J=1)) then
    U_ElemFxdFce_L[m,2] := -UDistLd[j].Uy_L*L[m]/2 +
                                U_ElemFxdFce_L[m,2];

(* moment at left hand end *)
If (S_J=0)and (E_J=1) then
    U_ElemFxdFce_L[m,3] := -UDistLd[j].Uy_L*sqr(L[m])/8 +
                                U_ElemFxdFce_L[m,3]
else
If (S_J=1)and (E_J=0) then
    U_ElemFxdFce_L[m,3] := U_ElemFxdFce_L[m,3]
else
If (S_J=0)and (E_J=0) then
    U_ElemFxdFce_L[m,3] := -UDistLd[j].Uy_L*sqr(L[m])/12 +
                                U_ElemFxdFce_L[m,3]
else
If (S_J=1)and (E_J=1) then
    U_ElemFxdFce_L[m,3] := U_ElemFxdFce_L[m,3];

(* axial force at right hand end *)
U_ElemFxdFce_L[m,4] := -UDistLd[j].Ux_L*L[m]/2 +
                                U_ElemFxdFce_L[m,4];

(* shear force at right hand end *)
If (S_J=0)and (E_J=1) then
    U_ElemFxdFce_L[m,5] := -3*UDistLd[j].Uy_L*L[m]/8 +
                                U_ElemFxdFce_L[m,5]
else
If (S_J=1)and (E_J=0) then
    U_ElemFxdFce_L[m,5] := -5*UDistLd[j].Uy_L*L[m]/8 +
                                U_ElemFxdFce_L[m,5]
else
If ((S_J=0)and (E_J=0)) OR ((S_J=1)and (E_J=1)) then
    U_ElemFxdFce_L[m,5] := -UDistLd[j].Uy_L*L[m]/2 +
                                U_ElemFxdFce_L[m,5];

(* moment at right hand end *)
If (S_J=0)and (E_J=1) then
    U_ElemFxdFce_L[m,6] := U_ElemFxdFce_L[m,6]
else
If (S_J=1)and (E_J=0) then
    U_ElemFxdFce_L[m,6] := UDistLd[j].Uy_L*sqr(L[m])/8 +
                                U_ElemFxdFce_L[m,6]
else
If (S_J=0)and (E_J=0) then
    U_ElemFxdFce_L[m,6] := UDistLd[j].Uy_L*sqr(L[m])/12 +
                                U_ElemFxdFce_L[m,6]
else
If (S_J=1)and (E_J=1) then
    U_ElemFxdFce_L[m,6] := U_ElemFxdFce_L[m,6];

END;      (*FOR j*)
Writeln(data1);
Writeln(data1,'Element Fixed End Forces in Local System by Uniform Loads only :');
Writeln(data1,'F"x@s':11,char(32),'F"y@s':11,char(32),'Mz@s':11,
            char(32),'F"x@e':11,char(32),'F"y@e':11,char(32),
            'Mz@e':11);

Writeln(data1,' fixed end forces for members ');
FOR j := 1 TO NoMems DO BEGIN
    Writeln(data1,'Member:',j:3);
    FOR k:= 1 TO 6 DO BEGIN
        write(data1,U_ElemFxdFce_L[j,k]:11:4, chr(32));
    END;      (*FOR k16*)
    Writeln(data1);
END;      (*FOR j1NoMems*)
Writeln(data1);

(* calculates the total Fixed end forces due to Point and uniform loads *)
(* using the same parameter ElemFxdFce_L[j,k] which earlier was used *)
(* only for point loads, now is used for the total fixed end forces. *)
FOR j := 1 TO NoMems DO
    FOR k:= 1 TO 6 DO BEGIN
        ElemFxdFce_L[j,k] := U_ElemFxdFce_L[j,k] + ElemFxdFce_L[j,k];
    END;
END;

```

```

END;
Writeln(data1);
Writeln(data1, 'Element Fixed End Forces in Local System by all loads:');
Writeln(data1, 'F"x@s":11,char(32), 'F"y@s":11,char(32), 'Mz@s":11,
char(32), 'F"x@e":11,char(32), 'F"y@e":11,char(32),
'Mz@e":11);

Writeln(data1, ' fixed end forces for members ');
FOR j := 1 TO NoMems DO BEGIN
  Writeln(data1, 'Member:', j:3);
  FOR k:= 1 TO 6 DO BEGIN
    write(data1, ElemFxdFce_L[j,k]:11:4, chr(32));
  END; (*FOR k16*)
  Writeln(data1);
END; (*FOR j1NoMems*)
Writeln(data1);
END; (*IF (NoUDistLd >0) *)
END; (*PROCEDURE*)

PROCEDURE DetermineGlobalFixedEndForces;
VAR
  (* transferring the FEF's form Local to Glob system *)
  m, j, k : LONGINT;
BEGIN
  (* At this stage the ElemFxdFce_L[j,k,m] carries the total ElementFixedEnd *)
  (* Forces on 'j' members due to entire different types of loading on
  structure. *)

  (* in local coordinates. *)
  FOR m := 1 TO NoMems DO Begin
    FOR j := 1 TO 6 DO
      FOR k := 1 TO 6 DO begin
        ElemFxdFce_G[m,j] := ElemTposTransMat[m,j,k]*ElemFxdFce_L[m,k] +
          ElemFxdFce_G[m,j];
      end;
    End;
  Writeln(data1);
  Writeln(data1);
  Writeln(data1, 'Element Fixed End Forces in Global System by all loads :');
  Writeln(data1, 'Fx@s":11,char(32), 'Fy@s":11,char(32), 'Mz@s":11,
char(32), 'Fx@e":11,char(32), 'Fy@e":11,char(32), 'Mz@e":11);
  Writeln(data1);
  Writeln(data1, ' Fixed End Forces for members:');
  FOR j := 1 TO NoMems DO BEGIN
    Writeln(data1, ' Member: ', j:3);
    FOR k:= 1 TO 6 DO BEGIN
      write(data1, ElemFxdFce_G[j,k]:11:4, chr(32));
    END; (*FOR k16*)
    Writeln(data1);
    Writeln(data1);
  END; (*FOR j1NoMems*)
  Writeln(data1);
END;

PROCEDURE DetermineEffectiveNodalForces;
VAR
  S, (*start node*)
  E, (*end node*)
  j, k, m, n: LONGINT;

BEGIN (* modified for non-consecutive nodes setting *)
  m :=0; n:=0; S:=0; E:=0; (*initialising temporary parameters*)

  FOR j := 1 TO NoMems DO BEGIN (*induced by member loads*)
    S:= Member[j].S_Node;
    E:= Member[j].E_Node;

    FOR k := 1 TO 3 DO BEGIN
      EffNodeFce[3*(S-1)+k] := -ElemFxdFce_G[j,k] + EffNodeFce[3*(S-1)+k];
    END; (*FOR k13 *)
    FOR k := 4 TO 6 DO BEGIN
      EffNodeFce[3*(E-1)+(k-3)] :=
        -ElemFxdFce_G[j,k] + EffNodeFce[3*(E-1)+(k-3)];
    END; (*FOR k46 *)
  END; (*for j1Nomems *)
  FOR j := 1 TO NoNdLd DO - BEGIN (*induced by node loads*)
    N:= NdLd[j].name;
    EffNodeFce[3*(N-1)+1] := NdLd[j].XForce + EffNodeFce[3*(N-1)+1];
  
```

```

      EffNodeFce[3*(N-1)+2] := NdLd[j].YForce + EffNodeFce[3*(N-1)+2];
      EffNodeFce[3*(N-1)+3] := NdLd[j].ZMoment+ EffNodeFce[3*(N-1)+3];
END; (*FOR j1NoNdLd *)
Writeln(datal,' Structure Effective Nodal Forces(RHS) {P} - {P''} :');
Writeln(datal,' Node',Chr(9),'Force':10);
n := -1;
FOR j := 1 TO 3*NoNodes DO BEGIN
  Inc(n);
  m := n Div 3 +1;
  writeln(datal,m:4, Chr(9),EffNodeFce[j]:12:5);
END; (*FOR j13*NoNodes*)
writeln(datal);
END;      (*PROCEDURE*)

PROCEDURE DetermineDegreesOfFreedom;
VAR
  n, j, jj, k1, m : LONGINT;

BEGIN
  (* dof : counter for degree of freedom *)
  (* fd_dof : counter for degree of free displacement *)
  (* pd_dof: counter for degree of presc. displacement *)
  k1 := 0;      (* counter for node position *)
  n := 0;
  FOR j := 1 TO NoNdRs DO BEGIN
    n := NdRs[j].name; (* Restrained node identified *)
    Writeln(datal,'Restraint set number =',j:3,', Located at node = ',n:3);

    k1 := (n-1)*3 +1;
    IF (NdRs[j].Xdir = 1) THEN BEGIN
      Inc(dof);
      MatPosDOF[dof] := k1;
      Write(datal,'Matrix Position DOF, k1 =':27, k1:3);
      IF NoPresLd<>0 THEN BEGIN
        For jj:=1 TO NoPresLd DO
          IF (P_D_Ld[jj].name = N) THEN BEGIN
            IF (P_D_Ld[jj].XDispl <> 0) THEN BEGIN
              Inc(pd_dof);
              PD_MatPosDOF[pd_dof] := k1;
              PD_NodeDOF^[pd_dof,1] := P_D_Ld[jj].XDispl;
              Write(datal,Chr(9),'Presc. displ. in Xdir(m) =',P_D_Ld[jj].Xdispl:6:3);
            END ELSE (*IF PD Does not belong to this dof *)
              IF (P_D_Ld[jj].XDispl = 0 )THEN BEGIN
                Inc(Fd_dof);
                FD_MatPosDOF[fd_dof] := k1;
              END;
            END ELSE BEGIN (*IF NoPresLd=<=>N *)
              Inc(Fd_dof);
              FD_MatPosDOF[fd_dof] := k1;
            END;
          END ELSE BEGIN (* IF NoPresLd0*)
            Inc(Fd_dof);
            FD_MatPosDOF[fd_dof] := k1;
          END;
        END ELSE (*IF Xdir1*)
          IF (NdRs[j].Xdir = 2) THEN BEGIN (*IF dof is a spring *)
            Inc(dof);
            MatPosDOF[dof] := k1;
            Inc(Fd_dof);
            FD_MatPosDOF[fd_dof] := k1;
            Writeln(datal,'Matrix Position DOF, k1 =':27, k1:3);
          END; (*IF Xdir2*)
        Writeln(datal);

        k1 := (n-1)*3 +2;
        IF (NdRs[j].Ydir = 1) THEN BEGIN
          Inc(dof);
          MatPosDOF[dof] := k1;
          Write(datal,'Matrix Position DOF, k1 =':27,k1:3);
          IF NoPresLd<>0 THEN BEGIN
            For jj:=1 TO NoPresLd DO
              IF (P_D_Ld[jj].name = N) THEN BEGIN
                IF (P_D_Ld[jj].YDispl <> 0) THEN BEGIN
                  Inc(pd_dof);
                  PD_MatPosDOF[pd_dof] := k1;
                  PD_NodeDOF^[pd_dof,1] := P_D_Ld[jj].YDispl;
                  Write(datal,Chr(9),'Presc. displ. in Ydir(m) =',P_D_Ld[jj].Ydispl:6:3);
                END
              END
            END
          END
        END
      END
    END
  END

```

```

        END ELSE
        IF (P_D_Ld[jj].YDispl = 0 )THEN BEGIN
            Inc(Fd_dof);
            FD_MatPosDOF[fd_dof]:= k1;
        END;
    End ELSE BEGIN                                (*IF NoPresLd=<>N *)
        Inc(Fd_dof);
        FD_MatPosDOF[fd_dof]:= k1;
    END;
    END ELSE BEGIN                                (* IF NoPresLd0*)
        Inc(Fd_dof);
        FD_MatPosDOF[fd_dof]:= k1;
    END;
    END ELSE                                     (*IF Ydir1*)
    IF (Ndrs[j].Ydir = 2) THEN BEGIN
        Inc(dof);
        MatPosDOF[dof] := k1;
        Inc(Fd_dof);
        FD_MatPosDOF[fd_dof]:= k1;
        Writeln(data1,'Matrix Position DOF, k1 =':27, k1:3);
    END; (*IF Ydir2*)
        Writeln(data1);

    k1 := (n-1)*3 +3;
    IF (Ndrs[j].Zrot = 1) THEN BEGIN
        Inc(dof);
        MatPosDOF[dof] := k1;
        Write(data1,'Matrix Position DOF, k1 =':27,k1:3);
        IF NoPresLd<>0 THEN BEGIN
            For jj:=1 TO NoPresLd DO
                IF (P_D_Ld[jj].name = N) THEN BEGIN
                    IF (P_D_Ld[jj].ZDispl <> 0) THEN BEGIN
                        Inc(pd_dof);
                        PD_MatPosDOF[pd_dof]:= k1;
                        PD_NodeDOF^[pd_dof,1]:= P_D_Ld[jj].ZDispl;
                        Write(data1,Chr(9),'Presc. displ. in Zdir(rad) =',P_D_Ld[jj].Zdispl:6:3);
                    END ELSE
                    IF (P_D_Ld[jj].ZDispl = 0 )THEN BEGIN
                        Inc(Fd_dof);
                        FD_MatPosDOF[fd_dof]:= k1;
                    END;
                END ELSE BEGIN                                (*IF NoPresLd=<>N *)
                    Inc(Fd_dof);
                    FD_MatPosDOF[fd_dof]:= k1;
                END;
            END ELSE BEGIN                                (* IF NoPresLd0*)(*IF PresLd EXIST *)
                Inc(Fd_dof);
                FD_MatPosDOF[fd_dof]:= k1;
            END;
        END ELSE                                     (*IF Zrot1*)
        IF (Ndrs[j].Zrot = 2) THEN BEGIN
            Inc(dof);
            MatPosDOF[dof] := k1;
            Inc(Fd_dof);
            FD_MatPosDOF[fd_dof]:= k1;
            Writeln(data1,'Matrix Position DOF, k1 =':27, k1:3);
        END; (*IF Zrot2*)
        Writeln(data1);
        Writeln(data1);
        END; (*FOR j1NoNodes*)

    writeln(data1,' Total Degrees of freedom =',dof:3);
    writeln(data1,' NOTE : NON-FIXED nodes are listed below. ');
    FOR j := 1 TO 3*NoNodes DO
        IF (MatPosDOF[j] > 0) THEN BEGIN
            Writeln(data1,'dof='':5,j:3,
                ', Position in the (GSM) Global Stiffness Matrix =':25,MatPosDOF[j]:3);
            END;(*FOR j*)
        Writeln(data1);
    END; (*PROCEDURE*)

PROCEDURE AssembleDOFStiffnessMatrices;
VAR
    j, k, m, n : LONGINT;
(1) BEGIN

    (* NodalPrescDisplacement *)

```

```

(* [DOF StifMat] * DOFvector - Nodal Force vector = Reaction in DOF *)
(* BEGIN      (* confirmation for correct data setting an enough entry *)
temporary blockage
{   Writeln('To include the Nodal Prescribed Displacement, particular Degree(s)',
        ' of Freedom should earlier have been considered as 'FREE(s)''',
        ' in correspond to RESTRAINT section, otherwise, they are not',
        ' accepted as P_Displacements. ');

    Writeln('If there is any Nodal Prescribed Displacement Load, has ',
        ' the concerned Degree(s) of Freedom been accepted ?');
    Writeln('Answer [y] for yes and [n] for no...');
{   Readln(DOFaccepted);
{   if ( DOFaccepted IN No ) then begin
        Writeln('Program is interrupted here. Define DOF(s) in the RESTRAINT',
        ' Section, then run the program again. ');
{   Readln; GOTO 100;
{   end;
{   END; (* question *)

(* determine Complete dof stiffness matrix by *)
(* Picking up the related members from GSMatrix *)
IF (dof >0) then BEGIN

    FOR j := 1 TO dof DO BEGIN
        FOR k := 1 TO dof DO BEGIN
            StiffMatrixDOF^[j,k] := CombinSoilStrucStifMat^[MatPosDOF[j],MatPosDOF[k]];
        END;      (*FOR k*)
    END;          (*FOR j*)

    Writeln(datal, ' Putting DOF matrix/vector in order... ');
    Writeln(datal, ' Complete DOF Stiffness Matrix : ');
    Writeln(datal);
    FOR j := 1 TO dof DO begin
        FOR k := 1 TO dof DO BEGIN
            Write(datal, StiffMatrixDOF^[j,k]:12:3);
        END;      (*FOR kldof*)
        Writeln(datal);
    end;          (*FOR jldof*)
    Writeln(datal);
END;          (* IF dof0 *)

(* determine fd_dof stiffness matrix concern NON-P_DISPLACEMENTS *)
IF fd_dof >0 then BEGIN

    FOR j := 1 TO fd_dof DO BEGIN
        FOR k := 1 TO fd_dof DO BEGIN
            FD_StiffMatrixDOF^[j,k] :=
                CombinSoilStrucStifMat^[FD_MatPosDOF[j], FD_MatPosDOF[k]];
        END;      (*FOR k*)
    END;          (*FOR j*)

    Writeln(datal, ' Putting FD_DOF matrix/vector in order... ');
    Writeln(datal, ' FD_DOF Stiffness Matrix : ');
    Writeln(datal);
    FOR j := 1 TO fd_dof DO begin
        FOR k := 1 TO fd_dof DO BEGIN
            Write(datal, FD_StiffMatrixDOF^[j,k]:12:3);
        END;      (*FOR k1fd_dof*)
        Writeln(datal);
    end;          (*FOR j1fd_dof*)
    Writeln(datal);
END;          (* if fd_dof0 *)

(* determine Coef_pd_dof stiffmatrix is COEF. FOR  NODE PRESC.DISPLACEMENTS *)
IF pd_dof >0 then BEGIN

    FOR j := 1 TO fd_dof DO BEGIN
        FOR k := 1 TO pd_dof DO BEGIN
            Coef_PD_StiffMatrixDOF^[j,k] :=
                CombinSoilStrucStifMat^[FD_MatPosDOF[j], PD_MatPosDOF[k]];
        END;      (*FOR k*)
    END;          (*FOR j*)

    Writeln(datal, ' Putting COEF_DOF matrix in order... ');
    Writeln(datal, ' COEF_DOF Stiffness Matrix : ');
    Writeln(datal);

```

```

FOR j := 1 TO fd_dof DO begin
  FOR k := 1 TO pd_dof DO BEGIN
    Write(data1,Coef_PD_StiffMatrixDOF^[j,k]:12:3);
    END;      (*FOR kldof*)
    Writeln(data1);
  end;      (*FOR jldof*)
  Writeln(data1);
END;      (* if pd_dof0 *)

FOR k := 1 TO fd_dof DO
FOR m := 1 TO 1 DO
FOR n := 1 TO pd_dof DO BEGIN
  PD_NodeFce^[k,m] := Coef_PD_StiffMatrixDOF^[k,n] * PD_NodeDOF^[n,m] +
                    PD_NodeFce^[k,m];
END;      (*FOR n*)

Writeln(data1,'Node Force vector made by Pres. Displ.');
```

FOR j := 1 TO fd\_dof DO BEGIN

Writeln(data1,'dof''pos in GSM=',FD\_MatPosDOF[j]:3,','

pd\_dof=',j:3,PD\_NodeFce^[j,1]:12:3);

END; (\*FOR j\*)

Writeln(data1);

(\* assemble FD\_DOF force vector \*)

FOR j := 1 TO fd\_dof DO BEGIN

NodeFceDOF^[j,1] := EffNodeFce[FD\_MatPosDOF[j]] - PD\_NodeFce^[j,1];

END; (\*FOR j\*)

Writeln(data1,' Total Nodal force (PD\_considered) DOF vector:');

FOR j := 1 TO fd\_dof DO BEGIN

Writeln(data1,'fd\_dof=',j:3,NodeFceDOF^[j,1]:12:3);

END; (\*FOR j\*)

Writeln(data1);

**END;** (\*PROCEDURE\*)

**PROCEDURE** DetermineDispDOF;

VAR

j, n, m : LONGINT;

BEGIN

(\*main form of of A\*X=B; where A is coef for variable; X is Var, B is constant\*)

(\*1st parameter A(ie:FD\_StiffMatrixDOF) is a matrix of actual size N\*M stored i\*)

(\*physical size of NP\*NP(LHS); 2nd parameterB(ie:NodeFceDOF^)of actual size 1\*M\*)

(\*stored in input matrix of N\*M containing(RHS) vector stored in an array of\*)

(\*NP\*MP.In output, first para is replaced by its matrix inverse and second par\*)

(\*is replaced by the corresponding set of solution vector.\*)

Writeln(data1,' RHS (correspond to NON\_ZERO DOF) resulted from Loads and

PDisplacement{b4 calcs:}');

FOR j := 1 TO fd\_dof DO begin

IF (MatPosDOF[j] <> 0) THEN

Writeln(data1,j:4,Chr(9),NodeFceDOF^[j,1]:12:5);

END; (\*FOR j\*)

(\*#####\*)

Gaussj(FD\_StiffMatrixDOF^,fd\_dof,3\*MaxNodes,NodeFceDOF^,1,3\*MaxNodes);

(\*#####\*)

Writeln(data1,'Matrix is being solved ...');

Writeln(data1);

FOR j := 1 TO 3\*NoNodes DO

IF (MatPosDOF[j]<>0) THEN BEGIN

NodeDefl[FD\_MatPosDOF[j]]:= NodeFceDOF^[j,1];

NodeDefl[PD\_MatPosDOF[j]]:= PD\_NodeDOF^[j,1];

END; (\*IF\*)

Writeln(data1,'Whole Struc Nodal Deflections resulted from Loads and

PDisplacement(after calcs:}');

Writeln(data4,'Whole Struc Nodal Deflections:');

n:=-1;

FOR j := 1 TO 3\*NoNodes DO BEGIN

Inc(n);

m := n Div 3+1;

Writeln(data1,m:4,Chr(9),NodeDefl[j]:12:5);

Writeln(data4,m:4,Chr(9),NodeDefl[j]:12:5);



```

END;      (*FOR j*)
(readln; )
Writeln(data1);
END;      (*PROCEDURE*)

PROCEDURE DetermineReactions;
VAR
  j, k, m, n, t : LONGINT;
BEGIN
  FOR j := 1 TO 3*NoNodes DO BEGIN
    FOR k := 1 TO 3*NoNodes DO BEGIN
      NodeReac[j] := GlobalStifMat^[j,k]*NodeDefl[k]+NodeReac[j];
      END;      (*FOR*)
      NodeReac[j] := -EffNodeFce[j] + NodeReac[j];
    END;      (*FOR*)

    Writeln(data1, 'REACTION AND DEFLECTION AT EACH SUPPORT NODE, in Global system');
    Writeln(data1, 'Horizontal Reaction Fx(kN), Vertical Reaction Fy(kN)',
      'Moment Reaction Mz(kN.m).');

    k:=0; t:=0;
    Writeln(data3, 'Reaction & Deflection @ support in Global system');
    Writeln(data1, 'Node', Char(9), ' Spring ', Char(32), ' NodeG_Deflection ',
      Char(32), ' NodeReaction');
    Writeln(data3, 'Node', Char(9), ' Spring', Char(9), ' G_Defl',
      Char(9), ' G_React');

    FOR j := 1 TO NoNodes DO BEGIN
      Inc(k);
      IF (NdRs[j].Xdir=2) or (NdRs[j].Xdir=0 ) THEN BEGIN
        NodeReac[k] := NdRs[j].SprX*NodeDefl[k]+NodeReac[k];
        Writeln(data1, j:4, 'x', Char(32), NdRs[j].SprX:10:5, Char(32),
          NodeDefl[k]:15:5, Char(32), NodeReac[k]:15:5);
        Writeln(data3, j:2, 'x', Char(9), NdRs[j].SprX:10:5, Char(9),
          NodeDefl[k]:10:5, Char(9), NodeReac[k]:10:5);
      END;      (*IF*)

      Inc(k);
      IF (NdRs[j].Ydir = 2) or (NdRs[j].Ydir=0 ) THEN BEGIN
        NodeReac[k] := NdRs[j].SprY*NodeDefl[k]+NodeReac[k];
        Writeln(data1, j:4, 'y', Char(32), NdRs[j].SprY:10:5, Char(32),
          NodeDefl[k]:15:5, Char(32), NodeReac[k]:15:5);
        Writeln(data3, j:2, 'y', Char(9), NdRs[j].SprY:10:5, Char(9),
          NodeDefl[k]:10:5, Char(9), NodeReac[k]:10:5);
      END;      (*IF*)

      Inc(k);
      IF (NdRs[j].Zrot = 2) or (NdRs[j].Zrot=0 ) THEN BEGIN
        NodeReac[k] := NdRs[j].SprZ*NodeDefl[k]+NodeReac[k];
        Writeln(data1, j:4, 'z', Char(32), NdRs[j].SprZ:10:5, Char(32),
          NodeDefl[k]:15:5, Char(32), NodeReac[k]:15:5);
        Writeln(data3, j:2, 'z', Char(9), NdRs[j].SprZ:10:5, Char(9),
          NodeDefl[k]:10:5, Char(9), NodeReac[k]:10:5);
      END;      (*IF*)
    END;      (*FOR j1!NoNodes*)

    Writeln(data1);
    Writeln(data1);
    Writeln(data1, 'Node', Chr(9), ' Reactions in Global System :');
    n := -1;
    FOR j := 1 TO 3*NoNodes DO BEGIN
      Inc(n);
      m := n Div 3 + 1;
      Writeln(data1, m:4, Chr(9), NodeReac[j]:12:5); END;      (*FOR*)
      Writeln(data1);
    END;      (*PROCEDURE*)

  PROCEDURE DetermineG_ElementDeflections;
  VAR
    S, E, j, k, m : LONGINT;
  BEGIN
    S:=0; E:=0;      (*initialising temporary parameters*)

    FOR j := 1 TO NoMems DO BEGIN
      S:= Member[j].S_Node;
      E:= Member[j].E_Node;
      FOR k := 1 TO 3 DO BEGIN
        G_DispElem[j,k] := NodeDefl[3*(S-1)+k];

```

```

END;      (*FOR k13*)

FOR k := 4 TO 6 DO BEGIN
    G_DisplElem[j,k] := NodeDefl[3*(E-1)+(k-3)];
END;      (*FOR k46*)
END;      (*FOR j1NoMems*)

Writeln(data1,'Nodal Global Deflection Vector for Elements:');
FOR j := 1 TO NoMems DO BEGIN
    Writeln(data1,'Element: ',j:3);
    For k :=1 TO 6 DO begin
        Write(data1,G_DisplElem[j,k]:10:5);
    End;
    Writeln(data1);
    Writeln(data1);
END;      (*FOR j1NoMems*)
Writeln(data1);
Writeln(data1);
END;      (*PROCEDURE*)

PROCEDURE DetermineL_ElementDeflections;
VAR
    j, k, m : LONGINT;
BEGIN
    FOR j := 1 TO NoMems DO
        FOR k := 1 TO 6 DO
            FOR m := 1 TO 6 DO BEGIN
                L_DisplElem[j,k] := ElemTransMat[j,k,m] * G_DisplElem[j,m] + L_DisplElem[j,k];
            END;      (* FOR mkj *)
        END;
    END;

    Writeln(data1,'Nodal Local Deflection Vector for Elements:');
    FOR j := 1 TO NoMems DO BEGIN
        writeln(data1,'Element: ',j:3);
        For k :=1 TO 6 DO begin
            Write(data1,L_DisplElem[j,k]:10:5);
        End;
        Writeln(data1);
        Writeln(data1);
    END;      (*FOR j1NoMems*)
    Writeln(data1);
    Writeln(data1);
END;      (*PROCEDURE*)

PROCEDURE DetermineElementInternalForces;
VAR
    j, k, m : LONGINT;
BEGIN
    FOR j := 1 TO NoMems DO
        FOR k := 1 TO 6 DO BEGIN
            FOR m := 1 TO 6 DO BEGIN
                ReacElem[j,k] := ElemL_StiffMat[j,k,m]*L_DisplElem[j,m] + ReacElem[j,k];
            END;      (*FOR m16*)
            ReacElem[j,k] := ReacElem[j,k] + ElemFxdFce_L[j,k];
        END;      (*FOR k16*)
    END;

    Writeln(data1,'Internal forces for elements( In local system) :');
    Writeln(data3,'Internal forces for elements( In local system) :');
    FOR j := 1 TO NoMems DO BEGIN
        Writeln(data1,'Element: ',j:3);
        Write(data3,'Elm:',j:3,chr(9));
        For k :=1 TO 6 DO begin
            Write(data1,ReacElem[j,k]:10:5,Char(9));
            Write(data3,ReacElem[j,k]:10:5,Char(9));
        End;
        Writeln(data1);
        Writeln(data1);
        Writeln(data3);
    END;      (*FOR*)
END;      (*PROCEDURE*)

{*****}
{ * MAIN PROGRAM *}
{*****}

BEGIN
    NewData;

```

```

clrscr;
{ ShowMeReminderNote;}
InitialiseGlobalVars;
InitialiseMatrixesMethodArrays;
Writeln('Type in the user main output file name.',
        ' Preferred format <FileName>.out');
Readln(OutFileStructure);
ASSIGN (data1,'c:\MyDocu-1\3tp\A_work\AllOuts\' + OutFileStructure);
{ ASSIGN (data1,'c:\My Documents\3tp\A_work\AllOuts\' + OutFileStructure);
  check for if the format of PATH is applicable)
ASSIGN (data3,OutFulCompar);
ASSIGN (data4,NodalG_DeflCompar);
PutHeading1(data1);
AppendFile(data3,OutFulCompar);
AppendFile(data4,NodalG_DeflCompar);
LoadDataFromInputFiles(Member,NoPresLd,NoNdLd,NoPtLd,NoUDistLd);
DecideSoilModel;
If (SelectedSoilModel=1) or (SelectedSoilModel=2) then
  Load3rd_DDDataFromInputFiles
else if (SelectedSoilModel=0) then
  NoSoilMessage;
AllocateMemberDataConstants;
DetermineElementTransMatrices;
DetermineLocalElementStiffnessMatrices;
DetermineElemG_StiffMat;
DetermineGlobalStiffnessMatrix;
Try :=1;
If SelectedSoilModel<> 0 Then begin
  DetermineSoilFlexibilityMatrix;
  DetermineSoilStiffnessMatrix; end;
DetermineCombinedMatrices;
DetermineLocalFixedEndForces;
DetermineGlobalFixedEndForces;
DetermineEffectiveNodalForces;
DetermineDegreesOfFreedom;
AssembleDOFStiffnessMatrices;
DetermineDispDOF;
DetermineReactions;
DetermineG_ElementDeflections;
DetermineL_ElementDeflections;
DetermineElementInternalForces;
StampFile(data1);
Writeln;
Writeln(data1,' THE END ');
DrawALine(data3,98);
DrawALine(data4,68);
Writeln('Thank you for using this program.');
```

{technical notes:

- 1- (%) this sign is the modification for soil case}  
(during the testing it is recommended to use \$\$+ stack directive, but then to be removed.
- 2- The source data file for 3D-extension should exist in this directory and any limitation for the number of frames should be make prior to running this program.)

```

Unit Build;

{$D+,S-,N+}      (*L+ for further links*)

INTERFACE

  USES
    Crt, Dos, Globals;

PROCEDURE NewData;
PROCEDURE DisposeDynamicVars;
PROCEDURE LoadDataFromInputFiles
  (Var Member: MemberType;
   Var NoPresLd,NoNdLd,NoPtLd,NoUDistLd: LongInt);
PROCEDURE DecideSoilModel;
PROCEDURE Load3rd_DDataFromInputFiles;
FUNCTION LeadingZero(w : Word) : String;
PROCEDURE StampFile(var F:text);
PROCEDURE PutHeading1(var F:text);
PROCEDURE PutHeading2(var F:text; S:string);
PROCEDURE PutUpdateHeading(var F:text);
PROCEDURE AppendFile(var F:text; S:string);
PROCEDURE DrawALine(var F:text; j:Integer );
PROCEDURE EndProgram;

IMPLEMENTATION

PROCEDURE NewData;
BEGIN
  New(GlobalStifMat);
  New(CombinSoilStrucStifMat);
  New(SoilFlexMat);
  New(NetSoilFlexMat);
  New(SoilStifMat);
  New(StiffMatrixDOF);
  New(FD_StiffMatrixDOF);
  New(Coef_PD_StiffMatrixDOF);
  New(NodeFceDOF);
  New(PD_NodeDOF);
  New(PD_NodeFce);
END; (*PROCEDURE*)

PROCEDURE DisposeDynamicVars;
Begin
  Dispose(GlobalStifMat);
  Dispose(CombinSoilStrucStifMat);
  Dispose(SoilFlexMat);
  Dispose(NetSoilFlexMat);
  Dispose(SoilStifMat);
  Dispose(StiffMatrixDOF);
  Dispose(FD_StiffMatrixDOF);
  Dispose(Coef_PD_StiffMatrixDOF);
  Dispose(NodeFceDOF);
  Dispose(PD_NodeDOF);
  Dispose(PD_NodeFce);
END; (*PROCEDURE*)

Function LeadingZero(w : Word) : String;
var
  s : String;
begin
  Str(w:0,s);
  if Length(s) = 1 then
    s := '0' + s;
  LeadingZero := s;
END;

PROCEDURE StampFile(var F:text);
Begin
  GetDate(y,mont,day,dow);
  GetTime(h,mint,s,hund);
  Write(F,'Day & Date: ', days[dow],', ', day:0, '/',mont:0, '/', y:0);
  Writeln(F,'. Time: ',LeadingZero(h),':',LeadingZero(mint));
END;

```

```

PROCEDURE PutHeading1(var F:text);
Begin
  Rewrite(F);
  StampFile(F);
  Writeln(datal,'Pascal FONT is NEW COURIER, any windows Output is needed to use ARIAL
FONT for linage purpose.');
```

Writeln(datal,'Output File Name: ',OutFileStructure);

Writeln(datal,'Supporting Units: Dos, Crt, Build, Globals, SoilMode.');

Writeln(datal,'This file is a complete listing of the structure data and output.');

Writeln(datal,'{\*      VERY IMPORTANT NOTE: GLOBAL COORDINATES SHOULD BE SELECTED SUCH

\*)');

Writeln(datal,'{\*                      THAT ALL THE STRUCUTRE STAYS IN THE FIRST QUARTER.

\*)');

**END;**

**PROCEDURE** PutHeading2(var F:text; S:string);

Begin

{writeln('this is tessing line 93 oct200'); readln;}

Rewrite(F);

{writeln('this is tessing line 95 oct200, then i halt it.');

readln; halt;}

Writeln(F,'Name of this file: ',S,'');

StampFile(F);

Writeln(F,' This file is a comparative results of different runs of analysis.');

Writeln(F,' A summary of multiple O/put file: ',OutFileStructure,' ',

'      3D-I/put file: ',SpaceData,'');

Writeln(F);

**END;**

**PROCEDURE** PutUpdateHeading(var F:text);

Begin

Writeln(F,'Data obtained from main O/put file: ',OutFileStructure,'');

Write(F,' Last Updated on ');

StampFile(F);

**END;**

**PROCEDURE** AppendFile(var F:text; S:string);

Begin

  {\$I-}

  Append(F);

  {\$I+}

  If IOResult <>0 Then PutHeading2(F,S);

  Writeln(F);

  DrawALine(F,50);

  PutUpdateHeading(F);

**END;**

**PROCEDURE** LoadDataFromInputFiles

  (Var Member: MemberType;

    Var NoPresLd,NoNdLd,NoPtLd,NoUDistLd: LongInt);

**PROCEDURE** GetNodeGeometry;

**PROCEDURE** ValidateItems;

  VAR

    k: LONGINT;

  BEGIN

    if NoNodes > 1 then Begin

      for k:= 1 to NoNodes do

        if Node[k].Name<>k then begin

          Writeln('Error in Sequence of Node Numbering');

          Close(datal); Writeln(Chr(7)); Readln;

          EndProgram; HALT;

        end;

      for k:= 1 to NoNodes-1 do begin

        if (Node[k].Xcoord=Node[k+1].Xcoord) then

        if (Node[k].Ycoord=Node[k+1].Ycoord) then begin

          Writeln('Error: At least two Nodes are overlapping.');

          Close(datal); Writeln(Chr(7)); Readln;

          EndProgram; HALT;

        end;{if Y}

      end;{for k}

    End;{if NoNodes1}

  END;{validation}

  VAR

    i, j       : LONGINT;

```

BEGIN
  ASSIGN(data, InputFileNodeGeom);
  RESET(data);
  Readln(data, JunkStr); Writeln(data1, JunkStr);
  Readln(data, JunkStr); Writeln(data1, JunkStr);
  Readln(data, NoNodes); Writeln(data1, NoNodes:3);

  if NoNodes = 1 then begin
    Writeln(data1, 'Structure does not have enough node. ');
    Writeln('Structure does not have enough node.',
      'Program is interrupted at this point.',
      ' retrieve the main data entry and restart. ');
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; HALT;
  end;

  if (NoNodes > MaxNodes) then begin
    Writeln('Number of Nodes is greater than MaxNodes.',
      'Program is interrupted at this point.',
      ' INCREASE MaxNodes in the CONST block',
      ' recompile the program and run it again. ');
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; HALT;
  end;

  Readln(data, JunkStr); Writeln(data1, JunkStr);
  Readln(data, JunkStr); Writeln(data1, JunkStr);

  for i:= 1 to NoNodes do
  with Node[i] do begin
    Readln(data, name, JunkChar, Xcoord, JunkChar, Ycoord, JunkChar, Descr);
    Write(data1, name:4, Chr(9), Xcoord:12:3, Chr(9), Ycoord:12:3, Chr(9));
    for j:= 1 TO 54 DO write(data1, Descr[j]);
    if (Descr[2] <> '') then Writeln(data1, '')
    else Writeln(data1);
  end;

  if not EOF(data) then begin
    Writeln('The input data file *.TGM contains more than initial declared',
      'items it needs to be checked before succeeding ',
      'the program. ');
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; HALT;
  end;
  ValidateItems;
  CLOSE(data);
  Writeln(data1);
  Writeln(data1);
END; (*procedure*)

PROCEDURE GetMatProperty;

PROCEDURE ValidateItems;
  VAR
    k: LONGINT;
  BEGIN
    if (NoMats =1) and (Mat[1].name =0) then begin
      Writeln('Error: No material type entered. ');
      Close(data1); Writeln(Chr(7)); Readln;
      EndProgram; HALT;
    end;

    if NoMats > 1 then Begin
      for k:= 1 to NoMats do
        if Mat[k].Name<>k then begin
          Writeln('Error in Sequence of Material Numbering');
          Close(data1); Writeln(Chr(7)); Readln;
          EndProgram; HALT;
        end;
      for k:= 1 to NoMats-1 do begin
        if (Mat[k].Elasticity_M=Mat[k+1].Elasticity_M) then
          if (Mat[k].Thermal_M=Mat[k+1].Thermal_M) then begin
            Writeln('Error: At least two Mat props are overlapping. ');
            Close(data1); Writeln(Chr(7)); Readln;
            EndProgram; HALT;
          end; {if Thermal}
        end; {for k}
      end;
    end;
  end;

```

```

End;{if}

END;{validation}

VAR
  i, j      : LONGINT;
BEGIN
  ASSIGN(data,InputFileMatProp);
  RESET(data);
  Readln(data,JunkStr); Writeln(data1,JunkStr);
  Readln(data,JunkStr); Writeln(data1,JunkStr);
  Readln(data,NoMats); Writeln(data1,NoMats:3);
  if (NoMats > MaxM_T) then begin
    Writeln('Size of the structure is bigger than expected',
      ', program will be interrupted at this point, ',
      ', please INCREASE THE MaxM_T in the CONST block',
      ', recompile the program then run it. ');
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; HALT;
  end;
  Readln(data,JunkStr); Writeln(data1,JunkStr);
  Readln(data,JunkStr); Writeln(data1,JunkStr);

  for i:= 1 to NoMats do
  with Mat[i] do begin
    Readln(data,name,JunkChar,Elasticity_M,JunkChar,Thermal_M,
      JunkChar,Descr);
    Write(data1,name:3,Chr(9),Elasticity_M:10:8,Chr(9),Thermal_M:8:5,
      Chr(9));
    {unit conversion(*1e6) from GPa => KPa will be done before calculations}
    for j :=1 TO 63 DO write(data1,Descr[j]);
    if (Descr[2] <> '') then Writeln(data1,'')
    else Writeln(data1);
  end;

  if not EOF(data) then begin
    Writeln('The input data file *.TPM contains more than initial declared',
      'items it needs to be checked before succeeding ',
      'the program. ');
    Close(data1); Writeln(Chr(7)); Readln; EndProgram; HALT;
  end;

  ValidateItems;
  CLOSE(data);
  Writeln(data1);
  Writeln(data1);
END; {*procedure*}

PROCEDURE GetSecProperty;

PROCEDURE ValidateItems;
VAR
  k : LONGINT;
Begin
  if (NoSecs =1) and (Sec[1].name=0) then begin
    Writeln('Error: No Cross Section type entered. ');
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; HALT;
  end;

  if NoSecs > 1 then Begin

    for k:= 1 to NoSecs do
      if (Sec[k].Name<>k) then begin
        Writeln('Error in Sequence of C_Section Numbering');
        Close(data1); Writeln(Chr(7)); Readln;
        EndProgram; HALT;
      end;
      for k :=1 to NoSecs-1 do
        if (Sec[k].Area=Sec[k+1].Area) then
          if (Sec[k].M_Inertia=Sec[k+1].M_Inertia) then
            if Sec[k].Width=Sec[k+1].Width then begin
              Writeln('Error: At least two Section props are overlapping. ');
              Close(data1); Writeln(Chr(7)); Readln;
              EndProgram; HALT;
            end;
      end;
    End;{if NoSecs1}

```

```

end;{validation}
VAR
  i, j      : LONGINT;
begin
  ASSIGN(data,InputFileSecProp);
  RESET(data);
  Readln(data,JunkStr); Writeln(data1,JunkStr);
  Readln(data,JunkStr); Writeln(data1,JunkStr);
  Readln(data,NoSecs);  Writeln(data1,NoSecs:3);

  if (NoSecs > MaxS_T) then begin
    Writeln('Size of the structure is bigger than expected',
           ', program will be interrupted at this point, ',
           'please INCREASE THE MAXCES_TYPE in the CONST block',
           ', recompile the program then run it. ');
    Writeln(Chr(7));
    Close(data1); Writeln(Chr(7)); Readln;
    HALT;
  end;

  Readln(data,JunkStr); Writeln(data1,JunkStr);
  Readln(data,JunkStr); Writeln(data1,JunkStr);

  for i:= 1 to NoSecs do
  with Sec[i] do begin
    Readln(data,name,JunkChar,Area,JunkChar,M_Inertia,
           JunkChar,Width,JunkChar,Descr);
    Write(data1,name:3,Chr(9),Area:10:5,Chr(9),M_Inertia:10:6,
          Chr(9),Width:8:3,JunkChar);
    for j :=1 TO 62 DO write(data1,Descr[j]);
    if (Descr[2] <> '') then Writeln(data1,'')
    else Writeln(data1);
  end;

  if not EOF(data) then begin
    Writeln('Input data file *.TCS contains more than initial declared',
           'items it needs to be checked before succeeding',
           'the program. ');
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; Halt;
  end;
  ValidateItems;
  CLOSE(data);
  Writeln(data1);
  Writeln(data1);
END; (*procedure*)

PROCEDURE GetMemberData;

PROCEDURE ValidateItems;
VAR
  k      : LONGINT;
Begin

  if (NoMems =1) and (Member[1].name=0) then begin
    Writeln('Error: No material type entered. ');
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; Halt;
  end;

  if NoMems >1 then begin

    For k:= 1 to NoMems do
      if Member[k].Name<>k then begin
        Writeln('Error in Sequence of Member Numbering');
        Close(data1); Writeln(Chr(7)); Readln;
        EndProgram; Halt;
      end;
    For k:= 1 to NoMems-1 do
      if (Member[k].S_Node=Member[k+1].S_Node) then
      if (Member[k].E_Node=Member[k+1].E_Node) then begin
        Writeln('Error: At least two Members are overlapping. ');
        Close(data1); Writeln(Chr(7)); Readln;
        EndProgram; Halt;
      end;
    end;{if NoMems1}
  end;{validation}

```



```

VAR
    j, k      : LONGINT;
    Fi_min    : extended;
BEGIN (1)
    ASSIGN(data, InputFileMemberData);
    RESET(data);
    Readln(data, JunkStr); Writeln(data1, JunkStr);
    Readln(data, JunkStr); Writeln(data1, JunkStr);
    Readln(data, NoMems); Writeln(data1, NoMems:3);
    Readln(data, JunkStr); Writeln(data1, JunkStr);

    if (NoMems > MaxMembers) then begin
        Writeln('Size of the structure is bigger than expected',
            ', program will be interrupted at this point. ',
            'please INCREASE THE MAXMEMBERS in the CONST block',
            ', recompile the program then run it. ');
        Writeln(Chr(7));
        Close(data1); Writeln(Chr(7)); Readln;
        EndProgram; Halt;
    end;

    Readln(data, JunkStr);
    Insert(' ', JunkStr, 1);
    Insert(' [CCW] ', JunkStr, 13);
    Writeln(data1, JunkStr);
    JunkStr35:= JunkStr;
    Writeln(data3, JunkStr35);
    Readln(data, JunkStr);
    Insert('Length[m]', JunkStr, 8); Insert('Fi[rad] ', JunkStr, 18);
    Writeln(data1, JunkStr);
    JunkStr35:= JunkStr;
    Writeln(data3, JunkStr35);

(2) For j:= 1 to NoMems do With Member[j] do Begin
    Readln(data, name, JunkChar, S_Node, JunkChar, E_Node, JunkChar,
        S_Join, JunkChar, E_Join, JunkChar,
        C_Sec, JunkChar, Mat, JunkChar, Descr);
    L:= SQRT(SQR(Node[E_Node].XCoord-Node[S_Node].XCoord)+
        SQR(Node[E_Node].YCoord-Node[S_Node].YCoord));

(3) if (L = 0) then begin
        writeln('Error in Member data, Due to Zero Element Length');
        Close(data1); Writeln(Chr(7)); Readln; EndProgram; Halt;
    end
(3) else begin { L<>0 lays in Y direction, Fi= +/- 0.5Pi }

        if (Node[E_Node].XCoord-Node[S_Node].XCoord)= 0 then begin
(4) if ((Node[E_Node].YCoord-Node[S_Node].YCoord) > 0) then
                Fi := 0.5*Pi
            else
                Fi := -0.5*Pi;
        end

(4) else begin( for general case: L<>0 does not lay in Y direction, Fi<> +/- .5Pi )
        Fi:= ArcTan((Node[E_Node].YCoord-Node[S_Node].YCoord)/
            (Node[E_Node].XCoord-Node[S_Node].XCoord));
        end;
(3) End;

    Write(data1, Name:5, Chr(32), L:8:3, Chr(9), Fi:8:5, Chr(32),
        S_Node:3, Chr(9), E_Node:3, Chr(9),
        S_Join:1, Chr(32), E_Join:4, Chr(9),
        C_Sec:7, Chr(9), Mat:8, Chr(9));
    Writeln(data3, Name:5, Chr(32), L:8:3, Chr(9), Fi:8:5, Chr(32), S_Node:3,
        Chr(9), E_Node:3);

    for k :=1 TO 56 DO write(data1, Descr[k]);
    if (Descr[2] <> '') then
        Writeln(data1, '')
    else
        Writeln(data1);

(2) End; {for-with}

    if not EOF(data) then begin
        Writeln('The input data file *.TMD contains more than initial declared',

```

```

        'items it needs to be checked before succeeding ',
        'the program.');
```

Close(data1); Writeln(Chr(7)); Readln;  
EndProgram; Halt;  
end;

Writeln(data3);  
ValidateItems;  
CLOSE(data);  
Writeln(data1);  
Writeln(data1, 'Note: Node and Member notations may have different settings.');

Writeln(data1);  
Writeln('Note: Node and Member notations may have different settings.');

Writeln('Keep in mind for some other possibilities...');  
Readln;  
**END;** (\*procedure\*)

**PROCEDURE** GetRestraint;  
**PROCEDURE** ValidateItems;  
Label  
11;  
VAR  
C\_M, { member counter are common in the external nodes}  
C\_M\_H, { member counter are common in the external nodes, hinged}  
k, m, ml,  
N, r : LONGINT;  
ItemNotValid : Boolean;  
begin  
C\_M:=0; C\_M\_H:=0; ml:=0; N:=0;

ItemNotValid := False;  
if (NoNdRs =1) and (NdRs[1].name=0) then ItemNotValid := True;  
if (ItemNotValid=True) then begin  
Writeln('Notice: No Node Restrain entered.');

Writeln('Notice: lack of restrain caused instability and need to',  
' halt the program');

Close(data1); Writeln(Chr(7));Readln;  
EndProgram; Halt;  
end;

ItemNotValid := True;  
for k:= 1 to NoNdRs do  
if (NdRs[k].name<>0) then ItemNotValid := False;  
if (ItemNotValid=True) then begin  
Writeln(data1, 'Notice: Restraint Data to anonymous Node is entered.');

Writeln('Notice: Anonymous Node is Restrained. Program is Halt.');

Close(data1); Writeln(Chr(7));Readln;  
EndProgram; Halt;  
end;

ItemNotValid := True;  
k:= 1;  
While ( (k<NoNdRs) and (ItemNotValid=True) ) do  
begin  
if (NdRs[k].Xdir<>1) then ItemNotValid := False;  
k:= k+1;  
end;

if (ItemNotValid=True) then begin  
Writeln(data1, 'Error: Instability in structure in X direction.');

Writeln('Notice: Program is interrupted (Instability in X).');

Close(data1); Writeln(Chr(7));Readln;  
EndProgram; Halt;  
end;

ItemNotValid := True;  
k:= 1;  
While ( (k<NoNdRs) and (ItemNotValid=True) ) do  
begin  
if (NdRs[k].Ydir<>1) then ItemNotValid := False;  
k := k+1;  
end;

if (ItemNotValid=True) then begin  
Writeln(data1, 'Error: Instability in structure in Y direction.');

Writeln('Notice: Program is interrupted (Instability in Y).');

Close(data1); Writeln(Chr(7));Readln;  
EndProgram; Halt;  
end;

```

ItemNotValid := True;
k:= 1;
While ( (k<NoNdRs) and (ItemNotValid=True) ) do
begin
  if( (NdRs[k].Zrot<>1) OR
    ((NdRs[k].Zrot=1) and ((NdRs[k].Xdir<>1) or (NdRs[k].Ydir<>1))) )then
    ItemNotValid := False;
    k :=k+1;
  end;
  if (ItemNotValid=True) then begin
    Writeln(datal,'Error: Instability in structure in Z direction. ');
    Writeln('Notice: Program is interrupted (Instability in Z). ');
    Close(datal); Writeln(Chr(7));Readln;
    EndProgram; Halt;
  end;
{temporary eliminated
  for k:= 1 to NoNdRs do
    if ((NdRs[k].Xdir=2) and (NdRs[k].SprX=0)) or
      ((NdRs[k].Ydir=2) and (NdRs[k].SprY=0)) or
      ((NdRs[k].Zrot=2) and (NdRs[k].SprZ=0)) then begin
      Writeln('Warning: A Semi Restrained Node with No External Spring Coefficient in
X, Y or Z direction exists. ');
      Writeln('Is there any elastic soil model applied to node ''',NdRs[k].name, ''
?');
      Repeat
        Write('Type [y] for yes or [n] for no. ');
        Readln(SoilModelApplied);
        If(SoilModelApplied in NO) then begin
          Writeln('Error: Instability at this node for Structure. Data is not
correct.',
            'Program is interrupted. Review the ''Node Restraint Data'');
          Writeln(Chr(7));Close(datal); Readln; EndProgram; Halt;
        end
        Until ((SoilModelApplied in Yes) or (SoilModelApplied in NO));
      end;
    }
  ItemNotValid := False;
  if (NoNdRs > 1) then
    for m:= 1 to (NoNdRs-1) do
      for k:= m+1 to (NoNdRs) do
        if NdRs[m].Name = NdRs[k].Name then begin
          ItemNotValid := True;
          Writeln('Error: At least two sets of Restrains are applied to one Node. ');
          Close(datal); Writeln(Chr(7)); Readln; EndProgram; Halt; end;

for r:=1 to NoNdRs do
  if (NdRs[r].Zrot =1) then begin      {determine the node's Zdir FREEDOM}
    N:= NdRs[r].name;

    for m:=1 to NoMems do
      if (Member[m].S_Node=N) then begin {being a member in common at start node}
        Inc(C_M); m1 :=m;
        if (Member[m].S_Join=1) then begin(having member a hinged start node)
          Inc(C_M_H); end; end
      else if (Member[m].E_Node=N) then begin{being a member in common at end node}
        Inc(C_M); m1 :=m;
        if (Member[m].E_Join=1) then begin {having member a hinged end node}
          Inc(C_M_H); end;
      end;

      if (C_M_H >= C_M) then begin
        Writeln('Trouble is in Node :',N:3);
        Writeln('In this structure, there is a node hinge_connected to members ',
          'which has caused a FREE ROTATION node and causes structure instability.',
          'Program is interrupted here, correct the data and run the program
again. ');
        Close(datal); Writeln(Chr(7)); Readln; EndProgram; Halt;
      end; {* if C_M2 *}

    end; {* for r1NoNdRs *}

  if (NoNdRs<>NoNodes) then begin
    Writeln('The number of Nodal Restrains does not match with the number of Nodes. ');
    Writeln('Program is interrupted here, correct the data and run the program again. ');

```

```

        Close(data1); Writeln(Chr(7)); Readln; EndProgram; Halt;
    end;

end;(validation)
VAR
    j, k      : LONGINT;
begin
    ASSIGN(data, InputFileRestrains);
    ASSIGN(data5, SupportData);
    RESET(data);
    RESET(data5);
    Readln(data5, JunkStr);
    Readln(data5, JunkStr);
    Readln(data5, JunkStr);
    Readln(data, JunkStr); Writeln(data1, JunkStr);
    Readln(data, JunkStr); Writeln(data1, JunkStr);
    Readln(data, NoNdRs); Writeln(data1, NoNdRs:3);

    if (NoNdRs <> NoNodes) then begin
        Writeln('Nodes number does not match the Nodes Restrains.',
            ' There is a restrain at a non-registered node.',
            ' Check with the structure data input :',
            ', a) Node Entry, b) Restrains Entry.',
            ', program will be interrupted at this point. ');
        Writeln(Chr(7));
        Close(data1); Writeln(Chr(7)); Readln;
        EndProgram; Halt;
    end;

    Readln(data, JunkStr); Writeln(data1, JunkStr);
    Readln(data, JunkStr); Writeln(data1, JunkStr);

    for j:= 1 to NoNdRs do
    with NdRs[j] do begin
        Readln(data, name, JunkChar, Xdir, JunkChar, SprX, JunkChar,
            Ydir, JunkChar, SprY, JunkChar,
            Zrot, JunkChar, SprZ, JunkChar, Descr);
        Write(data1, Name:4, Chr(9),
            Xdir:3, Chr(9), SprX:8:3, Chr(9),
            Ydir:3, Chr(9), SprY:8:3, Chr(9),
            Zrot:3, Chr(9), SprZ:8:3, Chr(9));
        for k :=1 TO 1 DO write(data1, Descr[k]);
        if (Descr[2] <> '') then Writeln(data1, '');
        else Writeln(data1, '');
    end;

    if not EOF(data) then begin
        Writeln('Input data file *.TRS contains more than initial declared',
            ' items. Check before succeed');
        Close(data1); Writeln(Chr(7)); Readln;
        EndProgram; Halt;
    end;

    k:=0; {Footings Nodes Data base is filled out.}
    writeln('In GetRestrains user''s comment interface is temporary blocked. '); readln;
    {temporary blockage}
    FOR j :=1 TO NoNdRs DO
    Begin
        Writeln('Is Node', NdRs[j].name:3, ' a footing node ? ');
        Repeat {this blocked section is replaced with reading from data5}
        { Write('Type [y] for yes and [n] for no. ');
          Readln(NodeOnSoil);
        }
        Readln(data5, NodeOnSoil);
        if (NodeOnSoil in Yes) then
        begin
            Inc(k);
            FtNd[k].Order := k;
            FtNd[k].Name := NdRs[j].Name;
        end;
        { Until ((NodeOnSoil in Yes) or (NodeOnSoil in No));
        }
    End; {for i:=1 to NoNdRs}

    IF (k <> NoFtNd) THEN
    BEGIN
        Writeln('Number of the nodes in contact with soil(supports) is not verified. ');
        Writeln('Program is in halt. Amend the data and run the program again. ');
        Readln; Close(data1); EndProgram; Halt;
    END

```

```

        END;
        ValidateItems;
        CLOSE(data);
        Writeln(data1);
        Writeln(data1);
    END; (*procedure*)

PROCEDURE GetPrescDisplLd;

    PROCEDURE ValidateItems(Var NoPresLd:LongInt);
    VAR
        m, n: LONGINT;
    begin
        if (NoPresLd = 1) and (P_D_Ld[1].name=1) then begin
            Writeln('JUST A REMINDER: ' 'No Prescribed Displacement entered' '.');
            NoPresLd := 0;
            Write('Hit ENTER to continue. '); Readln;
        end;

        if (NoPresLd > 1) then
            for n:= 1 to NoPresLd do
                if (P_D_Ld[n].Name = 0) then begin
                    Writeln('Notice: Non-Valid Prescribed Displ. Load entered. ');
                    Close(data1); Writeln(Chr(7)); Readln ;
                    EndProgram; Halt;
                end;

                if NoPresLd > 1 then
                    for n:= 1 to NoPresLd-1 do
                        if (P_D_Ld[n].Name = P_D_Ld[n+1].Name) then begin
                            Writeln('Error: At least two Prescribed Displ. Loads are on one',
                                ' Node. ');
                            Close(data1); Writeln(Chr(7)); Readln;
                            EndProgram; Halt;
                        end; {if *.name*}

                        if (NoPresLd > 1) then
                            for n:= 1 to NoPresLd do
                                with P_D_Ld[n] do
                                    if (Name <> 0) and (XDispl=0) and (YDispl=0) and (ZDispl=0) then begin
                                        Writeln('Notice: Non-Valid Prescribed Displ. Load entered. ');
                                        Close(data1); Writeln(Chr(7)); Readln ;
                                        EndProgram; Halt;
                                    end;
                                end;

                            if (NoNdRs<>NoNodes) then begin
                                Writeln('The number of restrains does not match with the number of nodes,',
                                    ' for each node there should be an individual restrain. Program is',
                                    ' interrupted here. Run the program after the correction. ');
                                Close(data1); Writeln(Chr(7)); Readln;
                                EndProgram; Halt;
                            end; {if * NoPresLd *}

                            for m:=1 to NoPresLd do begin
                                if (P_D_Ld[m].XDispl<>0) then (* Node under P_D_load in Xdirection *)
                                    if (NdRs[P_D_Ld[m].name].XDir <> 1) then begin
                                        Writeln('DOF does not have a correct corresponding Restrain mode. Xdir should be
FREE.(1) ',
                                            'Program is interrupted. Run the program after correction. ');
                                        Close(data1); Writeln(Chr(7)); Readln; EndProgram; Halt;
                                    end;
                                if (P_D_Ld[m].YDispl<>0) then (* Node under P_D_load in Ydirection *)
                                    if (NdRs[P_D_Ld[m].name].YDir <> 1) then begin
                                        Writeln('DOF does not have a correct corresponding Restrain mode. Ydir should be
FREE.(1) ',
                                            'Program is interrupted. Run the program after correction. ');
                                        Close(data1); Writeln(Chr(7)); Readln; EndProgram; Halt;
                                    end;
                                if (P_D_Ld[m].ZDispl<>0) then (* Node under P_D_load in Zdirection *)
                                    if (NdRs[P_D_Ld[m].name].Zrot <> 1) then begin
                                        Writeln('DOF does not have a correct corresponding Restrain mode. Zrot should be
FREE.(1) ',
                                            'Program is interrupted. Run the program after correction. ');
                                        Close(data1); Writeln(Chr(7)); Readln; EndProgram; Halt;
                                    end;
                                end;
                            end;

```

```

end;(validation)

VAR
  j, k      : LONGINT;
begin
  ASSIGN(data,InputFilePresLoad);
  RESET(data);
  Readln(data,JunkStr); Writeln(datal,JunkStr);
  Readln(data,JunkStr); Writeln(datal,JunkStr);
  Readln(data,NoPresLd);Writeln(datal,'Before evaluation:',NoPresLd:3);

  if (NoPresLd > MaxNodes) then begin
    Writeln('Number of Prescribed loads is greater than expected.',
      ' There is a Prescribed load at a non-registered node.',
      ' Check with the load data input.',
      ', a) Node Entry, b) Prescribed Load Entry.',
      ', Program will be interrupted at this point.');
```

Writeln(Chr(7));

Close(datal); Writeln(Chr(7)); Readln;

EndProgram; Halt;

end;

Readln(data,JunkStr); Writeln(datal,JunkStr);

Readln(data,JunkStr); Writeln(datal,JunkStr);

for j:= 1 to NoPresLd do

with P\_D\_Ld[j] do begin

Readln(data,Name,JunkChar,XDispl,JunkChar,YDispl,JunkChar,

ZDispl,JunkChar,Descr);

Write(datal,name:4,Chr(9),XDispl:6:3,Chr(9),YDispl:8:3,Chr(32),

ZDispl:10:4,Chr(9));

for k :=1 TO 1 DO write(datal,Descr[k]);

if (Descr[2] <> '') then Writeln(datal,'')

else Writeln(datal);

end;

if not EOF(data) then begin

Writeln('The input data file \*.TPD contains more than initial ',

'declared items. Check before succeed');

Close(datal); Writeln(Chr(7)); Readln ;

EndProgram; Halt;

end;

ValidateItems(NoPresLd);

Writeln(datal,'NoPresLd Load approved = ',NoPresLd);

CLOSE(data);

Writeln(datal);

Writeln(datal);

**END;** (\*procedure\*)

**PROCEDURE** GetNodalLd;

**PROCEDURE** ValidateItems;

VAR

m : LONGINT;

begin

if (NoNdLd =1) and (NdLd[1].name=0) then begin

Writeln('JUST A REMINDER: ''No Node Load entered.'');

NoNdLd:= 0;

Write('Hit ENTER to continue. '); Readln ;

end;

for m:= 1 to NoNdLd-1 do

if NdLd[m].Name = NdLd[m+1].Name then begin

Writeln('Error:At least two Node Loads are applied to one Node.');

Writeln(Chr(7)); Readln;

Close(datal); EndProgram; Halt;

end;

end;(validation)

VAR

j, k : LONGINT;

begin

ASSIGN(data,InputFileNodeLoad);

RESET(data);

Readln(data,JunkStr); Writeln(datal,JunkStr);

```

Readln(data,JunkStr); Writeln(data1,JunkStr);
Readln(data,NoNdLd); Writeln(data1,'Before evaluation:',NoNdLd:3);

if (NoNdLd > MaxNodes) then begin
  Writeln('Number of Node-loads is greater than expected.',
    ' There is a Nodal Load at a non-registered node.',
    ' Check with the structure data input :',
    ' , a) Node Entry, b) Node Load Entry.',
    ' , program will be interrupted at this point.');
```

Writeln(Chr(7));

Writeln(Chr(7)); Readln;

Close(data1); EndProgram; Halt;

end;

```

Readln(data,JunkStr);
  Insert(' Load ',JunkStr,1);
  Writeln(data1,JunkStr);
Readln(data,JunkStr);
  Insert('Number ',JunkStr,1);
  Writeln(data1,JunkStr);

for j:= 1 to NoNdLd do
with NdLd[j] do begin
  Readln(data,Name,JunkChar,XForce,JunkChar,YForce,JunkChar,ZMoment,
    JunkChar,Descr);
  Write(data1,J:4,Chr(9),name:4,Chr(9),XForce:9:3,Chr(9),YForce:9:3,
    Chr(9),ZMoment:9:3,JunkChar);
  for k :=1 TO 1 DO write(data1,Descr[k]);
  if (Descr[2] <> '') then Writeln(data1,'')
  else Writeln(data1);
end;

if not EOF(data) then begin
  Writeln('The input data file *.TNL contains more than initial declared',
    ' items. Check before succeed');
  Writeln(Chr(7)); Readln;
  Close(data1); EndProgram; Halt;
end;

ValidateItems;
Writeln(data1,'NodalLd approved : ',NoNdLd);
CLOSE(data);
Writeln(data1);
Writeln(data1);
END; (*procedure*)

PROCEDURE GetMPPointLd;
PROCEDURE ValidateItems;
VAR
  m : LONGINT;
begin

if (NoPtLd =1) and (PtLd[1].mem=0) then begin
  Writeln('JUST A REMINDER: ''No Member Point Load entered.'');
  Writeln(data1,' Notice: No Member Point Load entered.');
```

NoPtLd := 0;

Write('Hit ENTER to continue. '); Readln;

end;

```

if (NoPtLd > 1) then
for m:= 1 to NoPtLd do
if (PtLd[m].mem =0) then begin
  Writeln('Notice: Non-Valid Member Point Load entered.');
```

Close(data1); Writeln(Chr(7)); Readln ;

EndProgram; Halt;

end;

```

if NoPtLd > 1 then      (existing an acceptable entry)
for m:= 1 to NoPtLd-1 do
if PtLd[m].Mem=PtLd[m+1].Mem then
if (PtLd[m].XLoc = PtLd[m+1].XLoc) then begin
  Writeln('Error: At least two Point Loads applied to a member',
    'at the same point. Program is interrupted.');
```

Close(data1); Writeln(Chr(7)); Readln;

EndProgram; Halt;

end;(if)

```

end;(validation)
VAR
  j, k      : LONGINT;
begin
  ASSIGN(data, InputFileMpointLd);
  RESET(data);
  Readln(data, JunkStr); Writeln(data1, JunkStr);
  Readln(data, JunkStr); Writeln(data1, JunkStr);
  Readln(data, NoPtLd); Writeln(data1, 'Before evaluation:', NoPtLd:3);

  if (NoPtLd > MaxPtLd) then begin
    Writeln('Size of the structure is bigger than expected',
      ', program will be interrupted at this point, ',
      ', please INCREASE THE MAXPTLD in the CONST block',
      ', recompile the program then run it. ');
    Writeln(Chr(7));
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; Halt;
  end;

  Readln(data, JunkStr); Writeln(data1, JunkStr);
  Readln(data, JunkStr); Writeln(data1, JunkStr);

  for j:= 1 to NoPtLd do
  with PtLd[j] do begin
    Readln(data, Pos, JunkChar, Mem, JunkChar, XLoc, JunkChar,
      Fx, JunkChar, Fy, JunkChar, Mz, JunkChar, Descr);
    Write(data1, Pos:4, Chr(32), Mem:8, Chr(32), XLoc:9:3, Chr(32),
      Fx:9:3, Chr(32), Fy:12:3, Chr(32), Mz:13:3, Chr(9));
    for k :=1 TO 1 DO write(data1, Descr[k]);
    if (Descr[2] <> '') then Writeln(data1, '');
    else Writeln(data1);
  end;

  if not EOF(data) then begin
    Writeln('The input data file *.TPL contains more than initial ',
      'declared items. Check before succeed');
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; Halt;
  end;

  ValidateItems;
  Writeln(data1, 'NoPtLd approved = ', NoPtLd);
  CLOSE(data);
  Writeln(data1);
  Writeln(data1);
END; (*procedure*)

PROCEDURE GetMUnifDLd;
PROCEDURE ValidateItems;
VAR
  m      : LONGINT;
begin

  if (NoUDistLd =1) and (UDistLd[1].mem=0) then begin
    Writeln('JUST A REMINDER: ''No Member Uniform Load entered.'');
    Write('Hit ENTER to continue. '); Readln ;
    NoUDistLd := 0;
  end;

  if (NoUDistLd > 1) then
  for m:= 1 to NoUDistLd do
  if (UDistLd[m].mem =0) then begin
    Writeln('Notice: Non-Valid Member for Uniform Load entry. ');
    Close(data1); Writeln(Chr(7)); Readln ;
    EndProgram; Halt;
  end;

  if NoUDistLd > 1 then
  for m:= 1 to NoUDistLd do
  if UDistLd[m].Mem=UDistLd[m+1].Mem then begin
    Writeln('Error: At least two Identical Distributed Loads applied',
      ' to a member. ');
    Writeln(Chr(7)); Readln ;
    Close(data1); EndProgram; Halt;
  end;

```



```

end;(validation)
VAR
  j, k      : LONGINT;
begin
  ASSIGN(data,InputFileMMunifLd);
  RESET(data);
  Readln(data,JunkStr); Writeln(data1,JunkStr);
  Readln(data,JunkStr); Writeln(data1,JunkStr);
  Readln(data,NoUDistLd); Writeln(data1,'Before evaluation:',NoUDistLd:3);

  if (NoUDistLd > MaxUDLd) then begin
    Writeln('Size of the structure is bigger than expected',
      ', program will be interrupted at this point, ',
      'please INCREASE THE MAXUDLD in the CONST block',
      ', recompile the program then run it. ');
    Close(data1); Writeln(Chr(7)); Readln;
    EndProgram; Halt;
  end;

  Readln(data,JunkStr); Writeln(data1,JunkStr);
  Readln(data,JunkStr); Writeln(data1,JunkStr);

  for j:= 1 to NoUDistLd do
  with UDistLd[j] do begin
    Readln(data,Mem,JunkChar,Ux,JunkChar,Uy,JunkChar,Descr);
    Write(data1,Mem:4,Chr(9),Ux:9:3,Chr(9),Uy:9:3,Chr(9));
    for k :=1 TO 1 DO write(data1,Descr[k]);
    if (Descr[2] <> '') then Writeln(data1,'')
    else Writeln(data1);
  end;

  ValidateItems;
  Writeln(data1,'NoUDistLd approved = ',NoUDistLd);
  CLOSE(data);
  Writeln(data1);
  Writeln(data1);
END; (*procedure*)

PROCEDURE UpdateStructureSize;{%}
BEGIN
  Writeln('Following data has been suggested. ');
  Writeln('Number of Nodes in contact with soil=',NoFtNd:3);

  Writeln('(Analytical Method) ISOTROPIC SOIL: ');
  Writeln('Elastic Mod.(kPa)=',E_Soil:6,'; Poisson''s ratio= ',NuSoil:5:3);
  Writeln;
  Writeln('(Numerical Method) CROSS-ANISOTROPIC SOIL: ');
  Writeln('Elastic Mod.(kPa), Eh=',Eh:6,'; Ev=',Ev:6,'; Fv=',Fv:6,'; Poisson''s ratios:
Vvh=',Vvh:5:3,
      ', Vhv=',Vhv:5:3,'; Vh=',Vh:5:3);
  Writeln;
  Writeln('(Numerical Method) ISOTROPIC SOIL: ');
  Writeln('Young Modul(kPa)= ',YM:6:0,'; Poisson ratio PR=',PR:5:3);

  Repeat
    Write('If the above constants are correct press [y] to proceed, or [n] to quit.
');
    Readln(ConstantsAccepted);
    IF (ConstantsAccepted IN No ) THEN BEGIN
      Write('Program halted. Amend the constants (Build unit,CONST) then resume the
exec. ');
      Readln; Close(data1); Close(data3); Close(data4); EndProgram; Halt;
    END;{question1}
    Until ( (ConstantsAccepted IN No) or (ConstantsAccepted IN Yes));
  END;

BEGIN
  GetNodeGeometry;
  GetMatProperty;
  GetSecProperty;
  GetMemberData;
  GetRestraint;
  GetPrescDisplLd;
  GetNodalLd;
  GetMPointLd;
  GetMUnifDLd;
  UpdateStructureSize;

```

```

END;(*PROCEDURE*)

PROCEDURE DecideSoilModel;
BEGIN
  Writeln('Select soil model from the following three options:');
  Repeat
    Write('Type: [0] no soil model applies; [1] Isotropic; [2] Cross-anisotropic; ');
    Readln(SelectedSoilModel);
    If (SelectedSoilModel <>0) and (SelectedSoilModel <>1) and
      (SelectedSoilModel <>2) then
      Write('The selected option is INVALID. Type only: 0, 1 or 2 for your
selection...');
    Until (SelectedSoilModel =0) or (SelectedSoilModel =1) or
      (SelectedSoilModel =2);

    Write('Selected Soil Model= ');
    Write(data1,'Selected Soil Model= ');
    Write(data3,'Selected Soil Model= ');
    Write(data4,'Selected Soil Model= ');
    If SelectedSoilModel= 0      then begin
      Writeln(' No Soil Model applies. ');
      Writeln(data1,' No Soil Model applies. ');
      Writeln(data3,' No Soil Model applies. ');
      Writeln(data4,' No Soil Model applies. '); end
    Else If SelectedSoilModel= 1 then begin
      Writeln(' Isotropic Half Space. ');
      Writeln(data1,' Isotropic Half Space. ');
      Writeln(data3,' Isotropic Half Space. ');
      Writeln(data4,' Isotropic Half Space. '); end
    Else If SelectedSoilModel= 2 then begin
      Writeln(' Cross-anisotropic Half Space. ');
      Writeln(data1,' Cross-anisotropic Half Space. ');
      Writeln(data3,' Cross-anisotropic Half Space. ');
      Writeln(data4,' Cross-anisotropic Half Space. '); end;
  END;

PROCEDURE Load3rd_DDataFromInputFiles;
PROCEDURE ApplyExtension;
VAR
  j      : LONGINT;
Begin
  ASSIGN(data2,SpaceData);
  RESET(data2);
  Readln(data2,JunkStr);
  Insert('section of ',JunkStr,6);
  Writeln(data1,JunkStr);
  Writeln(data3,JunkStr);
  Readln(data2);(blank line (line 2)in the source file)

  Readln(data2,JunkStr);
  Writeln(data1,JunkStr); Writeln(data3,JunkStr);
  Readln(data2,BakNum);
  Writeln(data1,BakNum:3);Writeln(data3,BakNum:3);
  Readln(data2,JunkStr);
  Writeln(data1,JunkStr); Writeln(data3,JunkStr);
  Readln(data2,JunkStr);
  Writeln(data1,JunkStr); Writeln(data3,JunkStr);
  If BakNum>0 Then begin
    For j:=1 to BakNum do
      With BakFrame[j] do begin      {length of the junks needs to be confirmed}
        Readln(data2,FramNumber,JunkChar,RelDistanc,JunkChar,LoadFactor);
        Writeln(data1,FramNumber:3,Chr(9),RelDistanc:15:3,Chr(9),LoadFactor:13:3);
        Writeln(data3,FramNumber:3,Chr(9),RelDistanc:15:3,Chr(9),LoadFactor:13:3);
      end;
  end;

  Readln(data2);
  Writeln(data1);Writeln(data3);(broken spacer spacerline)

  Readln(data2,JunkStr);
  Writeln(data1,JunkStr); Writeln(data3,JunkStr);
  Readln(data2,ForNum);
  Writeln(data1,ForNum:3);Writeln(data3,ForNum:3);
  Readln(data2,JunkStr);
  Writeln(data1,JunkStr); Writeln(data3,JunkStr);
  Readln(data2,JunkStr);
  Writeln(data1,JunkStr); Writeln(data3,JunkStr);

```

```

    If ForNum>0 Then begin
        For j:=1 to ForNum do
            With ForFrame[j] do begin
                Readln(data2,FramNumber,JunkChar,RelDistanc,JunkChar,LoadFactor);
                Writeln(data1,FramNumber:3,Chr(9),RelDistanc:15:3,Chr(9),LoadFactor:13:3);
                Writeln(data3,FramNumber:3,Chr(9),RelDistanc:15:3,Chr(9),LoadFactor:13:3);
            end;
        end;
        Readln(data2,JunkStr);
        Writeln(data1,JunkStr); Writeln(data3,JunkStr);
        Writeln(data1); Writeln(data3);
        Close(data2);
    End; {procedure ApplyExtension}

    Begin
        Writeln('Is calculation Restricted to Diagonal Model(no interaction between soil
nodes)?');
        Repeat
            Write('Type [y] for yes or [n] for no. ');
            Readln(ConsiderAlone);
        Until (ConsiderAlone in NO) or (ConsiderAlone in YES);

        If (ConsiderAlone in NO) then begin
            Writeln('3D-Model applies?');
            Repeat
                Write('Type [y] for yes or [n] for no. ');
                Readln(ExtendedModelApplies);
            Until (ExtendedModelApplies in NO) or (ExtendedModelApplies in YES);
            If (ExtendedModelApplies in YES) then
                ApplyExtension;
            end else ExtendedModelApplies:='N';

        Begin
            Writeln(data1,'Number of nodes in contact with soil : ',NoFtNd:3,'');
            Writeln(data1,'Diagonal-alone approach accepted ? ',ConsiderAlone,chr(9),
                ' ; 3D_Model: ',ExtendedModelApplies);
            Writeln(data1);

            Writeln(data3,'Number of nodes in contact with soil : ',NoFtNd:3,'');
            Writeln(data3,'Diagonal-alone approach accepted ? ',ConsiderAlone,chr(9),
                ' ; 3D_Model: ',ExtendedModelApplies);

            Writeln(data4,'Number of nodes in contact with soil : ',NoFtNd:3,'');
            Writeln(data4,'Diagonal-alone approach accepted ? ',ConsiderAlone,chr(9),
                ' ; 3D_Model: ',ExtendedModelApplies);

        End;
    END; {PROCEDURE Load3rd_D}

    PROCEDURE DrawALine(var F:text; j:Integer );
    var i: Integer;
    Begin
        For i:=1 TO j Do
            Write(F,'-');
        Writeln(F);
    END;

    PROCEDURE EndProgram;

    (* {$I-} enable this compiler directive, avoids RT error at closing time if
the file is not open *)
    Begin
        {$I-} (* due to cases*)
        CLOSE(data1);
        CLOSE(data3);
        CLOSE(data4);
        CLOSE(data5);
        {$I+}
        DisposeDynamicVars;
        Writeln('***** THE END OF EXECUTION ***** ');
        Writeln('User''s main output file:',OutFileStructure);
        Readln;
    End;

    END. {unit build}

```

```

Unit SoilMode;

{$N+}      (*D+,N+ Do activate these when are necessary *)
INTERFACE

USES
  Crt, Dos, Globals, Build, SoilCoef;

VAR
  (* local to soil flexibility matrices *)
  i, m, mm,
  j, n, nn,
  k0, k    :LONGINT;

PROCEDURE NoSoilMessage;
PROCEDURE DetermineSoilFlexibilityMatrix;

IMPLEMENTATION

PROCEDURE NoSoilMessage;
Begin
  Writeln('Soil is not encountered in the analysis. Program ended.');
```

Writeln(data1,'Soil is not encountered in the analysis. Program ended.');

Writeln(data3,'Soil is not encountered in the analysis. Program ended.');

Writeln(data4,'Soil is not encountered in the analysis. Program ended.');

Readln;

EndProgram; Halt;

END;

```

PROCEDURE DetermineSoilFlexibilityMatrix;
label
  DoBosCerMethod,
  DoGerHarWarMethod;

PROCEDURE BFSpace (BFnum:LongInt; BFframe:FrameSetUpType;
  MatPos:integer; ActLength:Extended);
var
  BFi,
  BFj : LongInt;
  Interaction,(*effect of other nodes from the same frame or space frames*)
  ClearDistanc (*shortest distance between force point and displ. point*)
    : Extended;
begin
  If BFnum >0 Then
    For BFi :=1 to BFnum Do begin
      CumulDistanc:=0;
      ClearDistanc:=0;
      For BFj :=1 to BFi do
        CumulDistanc := CumulDistanc + BFframe[BFj].RelDistanc;
        ClearDistanc:= Sqrt(Sqr(CumulDistanc)+Sqr(ActLength));
        Case MatPos of
          1: Interaction:= CoefF11(ClearDistanc);
          2: begin if (m=n) and (mm=1) and (nn=2) then Interaction:= 0
              else Interaction:= CoefF12(ClearDistanc); end;
          3: Interaction:= CoefF13(ClearDistanc);
          4: begin if (m=n) and (mm=2) and (nn=1) then Interaction:= 0
              else Interaction:= CoefF21(ClearDistanc); end;
          5: Interaction:= CoefF22(ClearDistanc);
          6: begin if (m=n) and (mm=2) and (nn=3) then Interaction:= 0
              else Interaction:= CoefF23(ClearDistanc); end;
          7: Interaction:= CoefF31(ClearDistanc);
          8: begin if (m=n) and (mm=3) and (nn=2) then Interaction:= 0
              else Interaction:= CoefF32(ClearDistanc); end;
          9: Interaction:= CoefF33(ClearDistanc);
        end;
        SoilFlexMat^[i,j] := SoilFlexMat^[i,j] +
          BFframe[BFi].LoadFactor * Interaction;
      end;
    end;
  END; (*Function BFSpace*)

BEGIN (* PROCEDURE DetermineSoilFlexibilityMatrix *)
(*In this procedure, firstly, based on the Nodes in touch with soil, and the
following options, the General Soil Flexibility Matrix is built.
a) ConsiderdiagonalsAloneMod,
b) Only Original Frame,
c) Considering 3Dextension (fully populated),
```

Then only the applicable coef's of the matrix are selected.

EHS\_dor degrees(EHS degrees of Restrain) are taken for further calculations.\*)

i:=0; j:=0; m:=0; mm:=0; n:=0; nn:=0; k0:=0; k:=0;

```
(*analytic approach soil parameter evaluations*)
DoBosCerMethod: (* WARNING: label; Do not delete.*)
If (SelectedSoilModel=1) then begin (*isotropic*)
  Writeln(' (Analytic Mtd) ISOTROPIC SOIL is selected. ');
  Writeln('Elastic Mod.(kPa)=' ,E_Soil:6,'; Poisson''s ratio= ',NuSoil:5:3);
  Writeln(data1,'Elastic Mod.(kPa)=' ,E_Soil:6,'; Poisson''s ratio= ',NuSoil:5:3);
  Writeln(data3,'Elastic Mod.(kPa)=' ,E_Soil:6,'; Poisson''s ratio= ',NuSoil:5:3);
  Writeln(data4,'Elastic Mod.(kPa)=' ,E_Soil:6,'; Poisson''s ratio= ',NuSoil:5:3);
  Readln;

  CoefA:= (1- NuSoil*NuSoil) / (Pi*E_Soil); (*KPa*)
  CoefB:= (1+NuSoil)*(1-2*NuSoil) / (2*Pi*E_Soil);(*KPa*)
  CoefC:= (1+NuSoil) / (Pi*E_Soil);
  Writeln(data1,'CoefA=' ,CoefA,'; CoefB=' ,CoefB,'; CoefC=' ,CoefC);
end

(*numeric approach soil parameter evaluations*)
(*Eh, Vvh, Vvh, Vh are given as soil elastic constants*)
else if SelectedSoilModel=2 then begin (*cross-anisotropic*)
{Eh:= Int(Ehr); Fv:=Int(Fvr); in attempt to create integer form of these non-int.
quantities}
  Writeln(' (Numeric Mtd) CROSS-ANISOTROPIC SOIL: ');
  Writeln('Elastic Mod.(kPa), Eh=' ,Eh:6,'; Ev=' ,Ev:6,'; Fv=' ,Fv:6,'; Poisson''s
ratios: Vvh=' ,Vvh:5:3,
'; Vhv=' ,Vhv:5:3,'; Vh=' ,Vh:5:3);
  Writeln(data1,'Elastic Mod.(kPa), Eh=' ,Eh:6,'; Ev=' ,Ev:6,'; Fv=' ,Fv:6,'; Poisson''s
ratios: Vvh=' ,Vvh:5:3,
'; Vhv=' ,Vhv:5:3,'; Vh=' ,Vh:5:3);
  Writeln(data3,'Elastic Mod.(kPa), Eh=' ,Eh:6,'; Ev=' ,Ev:6,'; Fv=' ,Fv:6,'; Poisson''s
ratios: Vvh=' ,Vvh:5:3,
'; Vhv=' ,Vhv:5:3,'; Vh=' ,Vh:5:3);
  Writeln(data4,'Elastic Mod.(kPa), Eh=' ,Eh:6,'; Ev=' ,Ev:6,'; Fv=' ,Fv:6,'; Poisson''s
ratios: Vvh=' ,Vvh:5:3,
'; Vhv=' ,Vhv:5:3,'; Vh=' ,Vh:5:3);
  Readln;
  Aci := Eh*(1-Vhv*Vvh)/(1+Vh)/(1-Vh-2*Vhv*Vvh);
  Bci := Eh*(Vh+Vhv*Vvh)/(1+Vh)/(1-Vh-2*Vhv*Vvh);
  Cci := Eh*Vvh/(1-Vh-2*Vhv*Vvh);
  Dci := Eh*Vvh*(1-Vh)/Vhv/(1-Vh-2*Vhv*Vvh);
  Fci := Fv;

  if (Aci<=0) or (Dci<=0) or (Fci<=0) or (Aci*Aci<=Bci*Bci) or
(Dci*(Aci+Bci)<=2*Cci*Cci) or (Aci*Dci<=Cci*Cci)
or (1-Vh-2*Vhv*Vvh<=0) or (Eh<0) or (Ev<0) or (Fv<0) or (1-Vh<=0) or
(1+Vh<=0) then begin
    Writeln('At least one of the strain energy conditions is not satisfied',
'at S-379; Program is halted');
    Readln; EndProgram; Halt; end;
end;

If (SelectedSoilModel=2) then
BEGIN
  AlfSq := (Aci*Dci-Cci*Cci-Cci*Fci+Fci*SQRT(Aci*Dci))/(2*Fci*Dci);
  If (AlfSq<0) then begin
    Writeln('Error: Elastic constants of this soil does not satisfy require',
'ment of positive strain energy. '); readln;
    EndProgram; Halt; end
  else begin
    Alfa := SQRT(AlfSq); end;

  GamSq:= (Aci-Bci)/Fci;
  if (GamSq<=0) then begin
    Writeln('Error: Elastic constants of this soil does not satisfy require',
'ment of positive strain energy. '); readln;
    EndProgram; Halt; end
  else begin
    Gama := SQRT(GamSq); end;

  BetSq:= (Aci*Dci-Cci*Cci-Cci*Fci-Fci*SQRT(Aci*Dci))/(2*Fci*Dci);
  Writeln('AlfSq=' ,AlfSq,'; BetSq=' ,BetSq); readln;
  If (Alfa>=1-1E-4) and (Alfa<=1+1E-4) and (ABS(BetSq)<=1E-4) then
    (*considered alfa=1 & beta=0*)
```

```

Begin
  Writeln('Notice: Soil properties confirm isotropic condition.');
```

```

  Writeln(data1,'Notice: Soil properties confirm isotropic condition.');
```

```

  Writeln(data3,'Notice: Soil properties confirm isotropic condition.');
```

```

  Writeln(data4,'Notice: Soil properties confirm isotropic condition.');
```

```

  Repeat
    Writeln('Select one of the following approaches: ');
    Write('Type ''1'': analytic, or ''3'': numeric. ');read;
    Readln(SelectedSoilModel);

    if (SelectedSoilModel=3 ) then  Begin
      (*isotropic & numeric technique*)
      Writeln('The NUMERICAL approach is proceeded.');
```

```

      Writeln('(Numeric Mtd) ISOTROPIC SOIL:');
```

```

      Writeln('Young Modulus @ Poisson ration of Anal/Isotropic is used.');
```

```

      Writeln('Changes to these elastic variables may be required.');
```

```

      Writeln('Young Modulus(kPa)= ',YM:6:0,'; Poisson ratio PR=',PR:5:3);
      Readln;
      Writeln(data1,'The NUMERICAL approach is proceeded.');
```

```

      Writeln(data3,'The NUMERICAL approach is proceeded.');
```

```

      Writeln(data4,'The NUMERICAL approach is proceeded.');
```

```

      GoTo DoGerHarWarMethod;
      End

    else if (SelectedSoilModel=1) then  Begin
      Writeln('The ANALYTICAL approach is proceeded.');
```

```

      Writeln(data1,'The ANALYTICAL approach is proceeded.');
```

```

      Writeln(data3,'The ANALYTICAL approach is proceeded.');
```

```

      Writeln(data4,'The ANALYTICAL approach is proceeded.');
```

```

      Readln;
      GoTo DoBosCerMethod;  end;
    until (SelectedSoilModel=1) or (SelectedSoilModel=3);
  End  (*alfa=1 beta=0 decision making*)
  (*Variable "SelectedSoilModel" could have been changed in last block*)

  Else If (SelectedSoilModel=2) and ((Alfa<1-1E-4) or (Alfa>1+1E-4)) then
  BEgin (* considered alfa<>1 *)
    if (BetSq>1E-4) then begin
      (*case one= BetaSq significant positive*)
      Beta := SQRT(BetSq);
      BetaSqPos := True;
      BetaSqNeg := False;
      BetaZero := False;
      Writeln('General CISO case I; alfa<>1 & betaSq>0.');
```

```

      Writeln(data1,'General CISO case I; alfa<>1 & betaSq>0.');
```

```

      end
    else if (BetSq<-1E-4) then begin
      (*case two= BetaSq significant negative*)
      Beta := SQRT(-BetSq);
      BetaSqNeg := True;
      BetaSqPos := False;
      BetaZero := False;
      Writeln('General CISO case II; alfa<>1 & betaSq<0.');
```

```

      Writeln(data1,'General CISO case II; alfa<>1 & betaSq<0.');
```

```

      end
    else (*if (BetSq>-1E-3) and (BetSq<1E-3) then*) begin
      (*case three= beta zero*)
      Beta := 0;
      BetaZero := True;
      BetaSqPos := False;
      BetaSqNeg := False;
      Writeln('General CISO case III; alfa<>1 & betaSq=0.');
```

```

      Writeln(data1,'General CISO case III; alfa<>1 & betaSq=0.');
```

```

      end;
  End;
END;  (* if SelectedSoilModel=2*)

(* soil parameter evaluations for numerical method*)
DoGerHarWarMethod: (* WARNING: label; Do not delete.*)
  If (SelectedSoilModel=2) or (SelectedSoilModel=3) then
  BEgin
    IF (SelectedSoilModel=3) then begin
      if PR=0.5 then
        PR := 0.499;
      Aci := YM*(1-PR)/((1+PR)*(1-2*PR));
      Bci := YM*PR /((1+PR)*(1-2*PR));
      Cci := Bci;
    end
  End

```

```

Dci := Aci;
Fci := YM/(1+PR);
Alfa := 1.0;
Beta := 0.0;
Gama := 1.0;

S1 := 2*(1-SQR(PR))/YM;
S3 := (1+PR)*(1-2*PR)/YM;

T1 := (1+PR)*(1-2*PR)/YM;
T3 := 2*(1-SQR(PR))/YM;
T10 := 2*(1+PR)/YM;
end (*if selec..=3*)
ELSE BEGIN (*the below parameters are determined for soilmode=2*)
  IF (BetaZero <> True) then begin
    Rho := Alfa+Beta;
    Phi := Alfa-Beta;

    G1 := (2*Cci+Fci)*Rho*Phi*(1/Fci)*(1/(Rho-Phi))*(1/(Cci+Dci*SQR(Phi)));
    G2 := (2*Cci+Fci)*Rho*Phi*(1/Fci)*(1/(Rho-Phi))*(1/(Cci+Dci*SQR(Rho)));
    G3 := Rho*(2*Dci*SQR(Phi)-Fci)*(1/Fci)*(1/(Rho-Phi))*(1/(Cci+Dci*SQR(Phi)));
    G4 := Phi*(2*Dci*SQR(Rho)-Fci)*(1/Fci)*(1/(Rho-Phi))*(1/(Cci+Dci*SQR(Rho)));

    H1 := (2*Cci+Fci)*Phi*(1/Fci)*(1/(Rho-Phi))*(1/(Cci+Dci*SQR(Phi)));
    H2 := (2*Cci+Fci)*Rho*(1/Fci)*(1/(Rho-Phi))*(1/(Cci+Dci*SQR(Rho)));
    H3 := (2*Dci*SQR(Phi)-Fci)*(1/Fci)*(1/(Rho-Phi))*(1/(Cci+Dci*SQR(Phi)));
    H4 := (2*Dci*SQR(Rho)-Fci)*(1/Fci)*(1/(Rho-Phi))*(1/(Cci+Dci*SQR(Rho)));
    H10 := 2*(1/Fci)*(1/Gama);

    I1 := 2*Alfa*SQR(Aci*Dci)*(1/(Aci*Dci-SQR(Cci)));
    I3 := 1/(SQR(Aci*Dci)+Cci);

    J1 := 1/(SQR(Aci*Dci)+Cci);
    J3 := -2*Alfa*Dci*(1/(Aci*Dci-SQR(Cci)));
    J11 := 2*(1/Fci)*(1/Gama);
  end
  ELSE BEgin (*where BetaZero is True*)
    S1 := (2*Cci+Fci)*2*Dci*AlfSq*Alfa*(1/Fci)*(1/SQR(Cci+Dci*AlfSq));
    S3 := ((2*Cci+3*Fci-2*Dci*AlfSq)*Dci*AlfSq+Cci*Fci)*(1/Fci)*(1/SQR(Cci+Dci*AlfSq));

    T1 := (2*Cci+Fci)*(Dci*AlfSq-Cci)*(1/Fci)*(1/SQR(Cci+Dci*AlfSq));
    T3 := (2*Cci+Fci)*2*Dci*Alfa*(1/Fci)*(1/SQR(Cci+Dci*AlfSq));
    T10 := 2*(1/Fci)*(1/Gama);
  END; (*if BetaZero is True*)
end; (*else where selec..=2*)
END; (*if selec..=2 or 3*)

(*THE MAIN BODY OF THE FLEXIBILITY CALCULATION STARTS HERE*)

If (NoFtNd > NoNdrs) then begin
  Writeln('Footing Nodes number does not match the Nodes Restrains.',
    ' Check with the structure data input.',
    ', Program will be interrupted at this point.');
```

:

```

  Writeln(Chr(7)); Writeln(Chr(7)); Readln;
  EndProgram; Halt;
end;

(*assuming that node numbering is increasingly ordered from left to right*)
Begin
  k:=0;
  FOR j :=1 TO NoNdrs DO
    (*where there is a full restrain OR Soil modelled in any DOF*)
    IF Ndrs[j].Xdir <>1 then Inc(k)
    Else if Ndrs[j].Ydir <>1 then Inc(k)
    Else if Ndrs[j].Zrot <>1 then Inc(k);
  End;

  If (k < 3) then
  begin
    Writeln('Degrees of Restriction of support is not enough.(unstable structure)');
    Writeln('Program is in halt. Amend the data and run the program again.');
```

:

```

    Readln; EndProgram; Halt;
  end;

  If (SelectedSoilModel=1) then begin

```

```

        Writeln('Soil is isotropic; Analytic Mtd; Elastic Parameters: Nu=',NuSoil:5:3,',
Young Modulus=',E_Soil:7);
        Writeln(data1,'Soil is isotropic; Analytic Mtd; Elastic Parameters:
Nu=',NuSoil:5:3,',Young Mod[kPa]='',E_Soil:7);
        Writeln(data3,'Soil is isotropic; Analytic Mtd; Elastic Parameters:
Nu=',NuSoil:5:3,',Young Mod[kPa]='',E_Soil:7);
        Writeln(data4,'Soil is isotropic; Analytic Mtd; Elastic Parameters:
Nu=',NuSoil:5:3,',Young Mod[kPa]='',E_Soil:7);
    end
    Else If (SelectedSoilModel=2) then begin
        Writeln('Soil is cross-anisotropic; Numeric; Parameters: Eh=',Eh:7,
', Ev=',Ev:7,', Fv=',Fv:7,', Vvh=',Vvh:5:3,', Vh=',Vh:5:3,', Vhv=',Vhv:5:3);
        Writeln(data1,'Soil is cross-anisotropic; Numeric; [kPa]; Parameters: Eh=',Eh:7,
', Ev=',Ev:7,', Fv=',Fv:7,', Vvh=',Vvh:5:3,', Vh=',Vh:5:3,', Vhv=',Vhv:5:3);
        Writeln(data3,'Soil is cross-anisotropic; Numeric; [kPa]; Parameters: Eh=',Eh:7,
', Ev=',Ev:7,', Fv=',Fv:7,', Vvh=',Vvh:5:3,', Vh=',Vh:5:3,', Vhv=',Vhv:5:3);
        Writeln(data4,'Soil is cross-anisotropic; Numeric; [kPa]; Parameters: Eh=',Eh:7,
', Ev=',Ev:7,', Fv=',Fv:7,', Vvh=',Vvh:5:3,', Vh=',Vh:5:3,', Vhv=',Vhv:5:3);
    end
    Else If (SelectedSoilModel=3) then begin
        Writeln('Soil is isotropic; Numeric; Parameters: PR=',PR:5:3,', YM[kPa]='',YM:7:0);
        Writeln(data1,'Soil is isotropic; Numeric; PR=',PR:5:3,', YM[kPa]='',YM:7:0);
        Writeln(data3,'Soil is isotropic; Numeric; PR=',PR:5:3,', YM[kPa]='',YM:7:0);
        Writeln(data4,'Soil is isotropic; Numeric; PR=',PR:5:3,', YM=[kPa]',YM:7:0);
    end;

(*calculating the NUMBER OF bays(or soil members*)
{*****to be reviewed)
    Writeln(data1,'Fictitious Soil Members are as follows:');
    Writeln(data1,'FS_name', ' S_Node', ' E_Node', ' L[m] ');
    Writeln(data3,'Fictitious Soil Members are as follows:');
    Writeln(data3,'FS_name', ' S_Node', ' E_Node', ' L[m] ');
    FOR j:= 1 TO NoFtNd-1 DO
    WITH SoilMem[j] DO
    begin
        Sx := Node[FtNd[j].Name].XCoord;
        Ex := Node[FtNd[j+1].Name].XCoord;
        Length := Ex - Sx;
        Writeln(data1,j:4,FtNd[j].Name:8,Char(32),FtNd[j+1].Name:6,Char(32),Length:10:3);
        Writeln(data3,j:4,FtNd[j].Name:8,Char(32),FtNd[j+1].Name:6,Char(32),Length:10:3);
    end;
    Writeln(data1);
    Writeln(data3);

(*MINIMUM DISTANCE IS THE MAXIMUM SIZE OF EACH FOOTING.*)
(*THAT IS THE SIDE LENGTH FOR SQUARE case, AND RADIUS FOR CIRCULAR FOOTINGS.*)
(*THE EVALUATION OF Rmin should be CORRECTED.*)
(**** WARNING ****)
(* Rmin := SoilMem[1].Length;
FOR j := 2 TO NoFtNd-1 DO
    IF (Rmin >= SoilMem[j].Length) THEN Rmin := SoilMem[j].Length;

(* based on the theory assumption made for defl. at point of load*)
(*calculating the flexibility coef's*)

(*IMPORTANT NOTE: By choosing the boundry of '3xNoftNd'i am filling up
the soil flexi matrix by the nodes that are only in footing(no hole is
left in the matrix, however,the maximum size of matrix would still be
'MaxNodes' for a fully nodes supported (ie: beam structure).*)

    Writeln(data1,'Minimum dimension for footing, Rmin=',Rmin:5);

    FOR i := 1 TO 3*NoFtNd DO BEGIN (*block get-to-gether 2*)
        (*make the Rmin effect due to the soil memlength...*)
        m:= (i-1) DIV 3;
        FOR j := 1 TO 3*NoFtNd DO BEGIN (*3*)
            (*make the Rmin effect due to the soil memlength...*)
            n:= (j-1) DIV 3;
            mm := i - 3*m;
            nn := j - 3*n;

            IF (mm=1) and (nn=1) THEN MatPos:= 1
            ELSE IF (mm=1) and (nn=2) THEN MatPos:= 2
            ELSE IF (mm=1) and (nn=3) THEN MatPos:= 3
            ELSE IF (mm=2) and (nn=1) THEN MatPos:= 4
            ELSE IF (mm=2) and (nn=2) THEN MatPos:= 5
            ELSE IF (mm=2) and (nn=3) THEN MatPos:= 6

```



```

ELSE IF (mm=3) and (nn=1) THEN MatPos:= 7
ELSE IF (mm=3) and (nn=2) THEN MatPos:= 8
ELSE IF (mm=3) and (nn=3) THEN MatPos:= 9;

(*non-diagonal coef's of diagonal nodes remain zero as initialised*)
IF ( m=n) THEN BEGIN      (*diagonal blocks*)
  IF (MatPos=1) THEN BEGIN  (*D11*)
    SoilFlexMat^[i,j] := CoefD11(Rmin); (*original frame cases 1&3*)
    If (ConsiderAlone in NO)and(ExtendedModelApplies in YES)then
      begin
        BFSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSpace (ForNum, ForFrame, MatPos, ActLength);
      end;
    End      (*D11*)

  ELSE IF (MatPos=2) THEN Begin  (*D12*)
    SoilFlexMat^[i,j] := CoefD12(Rmin);
    If (ConsiderAlone in NO)and(ExtendedModelApplies in YES)then
      begin
        BFSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSpace (ForNum, ForFrame, MatPos, ActLength);
      end;
    End      (*D12*)

  ELSE IF (MatPos=3) THEN Begin  (*D13*)
    SoilFlexMat^[i,j] := CoefD13(Rmin);
    If (ConsiderAlone in NO)and(ExtendedModelApplies in YES)then
      begin
        BFSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSpace (ForNum, ForFrame, MatPos, ActLength);
      end;
    End      (*D13*)

  ELSE IF (MatPos=4) THEN Begin  (*D21*)
    SoilFlexMat^[i,j] := CoefD21(Rmin);
    If (ConsiderAlone in NO)and(ExtendedModelApplies in YES)then
      begin
        BFSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSpace (ForNum, ForFrame, MatPos, ActLength);
      end;
    End      (*D21*)

  ELSE IF (MatPos=5) THEN Begin  (*D22*)
    SoilFlexMat^[i,j] := CoefD22(Rmin); (*for original frame*)
    If (ConsiderAlone in NO)and(ExtendedModelApplies in YES)then
      begin
        BFSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSpace (ForNum, ForFrame, MatPos, ActLength);
      end;
    End      (*D22*)

  ELSE IF (MatPos=6) THEN Begin  (*D23*)
    SoilFlexMat^[i,j] := CoefD23(Rmin);
    If (ConsiderAlone in NO)and(ExtendedModelApplies in YES)then
      begin
        BFSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSpace (ForNum, ForFrame, MatPos, ActLength);
      end;
    End      (*D23*)

  ELSE IF (MatPos=7) THEN Begin  (*D31*)
    SoilFlexMat^[i,j] := CoefD31(Rmin);
    If (ConsiderAlone in NO)and(ExtendedModelApplies in YES)then
      begin
        BFSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSpace (ForNum, ForFrame, MatPos, ActLength);
      end;
    End      (*D31*)

  ELSE IF (MatPos=8) THEN Begin  (*32*)
    SoilFlexMat^[i,j] := CoefD32(Rmin);
    If (ConsiderAlone in NO)and(ExtendedModelApplies in YES)then
      begin
        BFSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSpace (ForNum, ForFrame, MatPos, ActLength);
      end;
    End      (*D32*)

```

```

ELSE IF (MatPos=9) THEN Begin (*D33*)
    SoilFlexMat^[i,j] := CoefD33(Rmin); (*for original frame*)
    If (ExtendedModelApplies in YES)and(ConsiderAlone in NO) then
    begin
        BFSSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSSpace (ForNum, ForFrame, MatPos, ActLength);
    end; (*if yes*)
End; (*D33*)
END (*IF m=n*)

ELSE IF (n>m) and (ConsiderAlone in NO) THEN BEGIN
(*non-diagonal Nodes considered 4*)
FOR k:= m+1 TO n DO
    ActLength := ActLength + SoilMem[k].Length;

IF (MatPos=1) THEN Begin (*F11*)
    SoilFlexMat^[i,j] := CoefF11(ActLength); (*F11 for original frame*)
    If (ExtendedModelApplies in YES) then
    begin
        BFSSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSSpace (ForNum, ForFrame, MatPos, ActLength);
    end; (*if yes*)
End (*F11*)

ELSE IF (MatPos=2) THEN Begin (*F12*)
    SoilFlexMat^[i,j] := CoefF12(ActLength); (*F12 for original frame*)
    If (ExtendedModelApplies in YES) then
    begin
        BFSSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSSpace (ForNum, ForFrame, MatPos, ActLength);
    end; (*if yes*)
End (*F12*)

ELSE IF (MatPos=3) THEN Begin
    SoilFlexMat^[i,j] := CoefF13(ActLength);(*F13 for original frame*)
    If (ExtendedModelApplies in YES) then
    begin
        BFSSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSSpace (ForNum, ForFrame, MatPos, ActLength);
    end; (*if yes*)
End (*F13*)

ELSE IF (MatPos=4) THEN Begin
    SoilFlexMat^[i,j] := CoefF21(ActLength); (*F21 for original frame*)
    If (ExtendedModelApplies in YES) then
    begin
        BFSSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSSpace (ForNum, ForFrame, MatPos, ActLength);
    end; (*if yes*)
End (*F21*)

ELSE IF (MatPos=5) THEN Begin
    SoilFlexMat^[i,j] := CoefF22(ActLength); (*F22 for original frame*)
    If (ExtendedModelApplies in YES) then
    begin
        BFSSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSSpace (ForNum, ForFrame, MatPos, ActLength);
    end; (*if yes*)
End (*F22*)

ELSE IF (MatPos=6) THEN Begin
    SoilFlexMat^[i,j] := CoefF23(ActLength); (*F23 for original frame*)
    If (ExtendedModelApplies in YES) then
    begin
        BFSSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSSpace (ForNum, ForFrame, MatPos, ActLength);
    end; (*if yes*)
End (*F23*)

ELSE IF (MatPos=7) THEN Begin
    SoilFlexMat^[i,j] := CoefF31(ActLength); (*F31 for original frame*)
    If (ExtendedModelApplies in YES) then
    begin
        BFSSpace (BakNum, BakFrame, MatPos, ActLength);
        BFSSpace (ForNum, ForFrame, MatPos, ActLength);
    end; (*if yes*)

```

```

End      (*F31*)

ELSE IF (MatPos=8) THEN Begin
  SoilFlexMat^[i,j] := CoefF32(ActLength); (*F32 for original frame*)
  If (ExtendedModelApplies in YES) then
    begin
      BFSpace (BakNum, BakFrame, MatPos, ActLength);
      BFSpace (ForNum, ForFrame, MatPos, ActLength);
    end;    (*if yes*)
  End      (*F32*)

ELSE IF (MatPos=9) THEN Begin
  SoilFlexMat^[i,j] := CoefF33(ActLength); (*F33 For original frame*)
  If (ExtendedModelApplies in YES) then
    begin
      BFSpace (BakNum, BakFrame, MatPos, ActLength);
      BFSpace (ForNum, ForFrame, MatPos, ActLength);
    end;    (*if yes*)
  End;      (*F33*)

  ActLength:= 0;
  END;      (*IF (m<n)***)
  END;      (*FOR j!1-3*NoFtNd 3*)
  END;      (*FOR i!1-3*NoFtNd block get-to-gether 2*)

(* the top section of the flexi matrix before refilling the bottom half *)
  Writeln(data1);
  Writeln(data1,'S-811; Note that the non-diagonal nodes are also included. ');
  Writeln(data1,'Top section of the Fully Populated Soil Flexibility, before',
    ' refilling the bottom half. ');

  FOR i := 1 TO 3*NoFtNd DO BEGIN
    FOR j := 1 TO 3*NoFtNd DO BEGIN
      Write(data1,SoilFlexMat^[i,j]:12:8,chr(9));
    END;    (*FOR j!13NoFtNd*)
    Writeln(data1);
    Writeln(data1);
  END;      (*FOR i!NoFtNd*)
  Writeln(data1);

(*rem: filling the second (bottom) half of the flexi matrix *)
  FOR i := 1 TO 3*NoFtNd DO
    FOR j := 1 TO 3*NoFtNd DO BEGIN
      IF (i>j) THEN
        SoilFlexMat^[i,j]:=SoilFlexMat^[j,i];
    END;

    Writeln(data1);
    Writeln(data1,'Note that the non-diagonal nodes are also included. ');
    Writeln(data1,'Fully Populated (GENERAL) Soil Flexibility Coef''s Matrix: ');

    FOR i := 1 TO 3*NoFtNd DO BEGIN
      FOR j := 1 TO 3*NoFtNd DO BEGIN
        Write(data1,SoilFlexMat^[i,j]:12:8,chr(9));
      END;    (*FOR j!13NoFtNd*)
      Writeln(data1);
      Writeln(data1);
    END;      (*FOR i!NoFtNd*)
    Writeln(data1);

(* Task is to identify the ELASTIC HALF SPACE 'Degrees Of Restrain "EHS-DOR"
(APPLICABLE) from the whole populated GENERAL flex matrix *)
  N :=0; EHS_dor:=0; k0:=0; k:= 0; (*REINITIALIZATION the variables*)

  For j:=1 to NoFtNd DO Begin
    N := FtNd[j].name;
    k := (j -1)*3 +1; (*K IS COLUMN & ROW COUNTER of fully pop soil matrix*)
    k0:= (N -1)*3 +1; (*K0 IS COLUMN & ROW COUNTER of glob structure matrix*)
    if ((NdRs[N].Xdir =2) and (NdRs[N].SprX=0)) then begin
      Inc(EHS_dor);
      (*register acceptable position of EHS in XDir*)
      SoilPosDOR[EHS_dor].FulPopSoil:= k;
      (*register acceptable position of EHS; XDir Glob struct matrix*)
      SoilPosDOR[EHS_dor].GlobStif:= k0;
    end;

    k := (j -1)*3 +2; (*Inc(k);*)
    k0:= (N -1)*3 +2; (*Inc(k0);*)

```

```

if ((NdRs[N].Ydir =2) and (NdRs[N].SprY=0)) then begin
(*if ((NodeReac[k0]<0) AND (TRY <=5)) then begin*)
  Inc(EHS_dor);
  SoilPosDOR[EHS_dor].FulPopSoil:= k;
  (*register acceptable position of EHS in YDir*)
  SoilPosDOR[EHS_dor].GlobStif := k0;
  (*register acceptable position of EHS; YDir Glob struct matrix*)
end;

k := (j -1)*3 +3;  (*Inc(k);*)
k0:= (N -1)*3 +3;  (*Inc(k0);*)
if ((NdRs[N].Zrot =2) and (NdRs[N].SprZ=0)) then begin
  Inc(EHS_dor);
  SoilPosDOR[EHS_dor].FulPopSoil:= k;
  (*register acceptable position of EHS in ZDir*)
  SoilPosDOR[EHS_dor].GlobStif := k0;
  (*register acceptable position of EHS; ZDir Glob struct matrix*)
end;
End; (*for j!NoFtNd*)

For i:= 1 to EHS_dor do begin
  For j :=1 to EHS_dor do begin
    NetSoilFlexMat^[i,j] :=
      SoilFlexMat^[SoilPosDOR[i].FulPopSoil,SoilPosDOR[j].FulPopSoil];
  end;
end;
Writeln(data1,
'(Net) Soil Flexibility Coef''s for the RESTRAINED NODES (main frame):');
FOR i := 1 TO EHS_dor DO BEGIN
  FOR j := 1 TO EHS_dor DO BEGIN
    Write(data1,NetSoilFlexMat^[i,j]:12:8,chr(9));
  END;  (*FOR j!13NoFtNd*)
  Writeln(data1);
  Writeln(data1);
END;    (*FOR i!NoFtNd*)
Writeln(data1);
END;  (*PROCEDURE FlexiMatrixIsotropicSoil*)
END.  (*unit build*)

```

# Appendix A5

# A5.1

Tables associated with  
Graphs 5.1-5.18  
Output of structural analysis  
associated with  
isotropic soil  
(sub-cases 1 & 2 from Table 5.1)

**Bending Moment ( $M_{32}$  and  $M_{34}$ ) vs Isotropic soil Modulus of Elasticity ( $E_H$ )**  
(See Table 5.1a)  $\nu=0.25$

item	E soil	D group	P group	S group
1-Iso	5	13.8708	13.8171	12.1923
2-Iso	10	18.2303	18.2473	17.1872
3-Iso	15	20.2241	20.2531	19.4632
4-Iso	20	21.3693	21.3996	20.7698
5-Iso	30	22.6349	22.6619	22.2136
6-Iso	40	23.3191	23.3423	22.9942
7-Iso	50	23.7478	23.7679	23.4834
8-Iso	70	24.2556	24.2712	24.0629
9-Iso	120	24.8058	24.8158	24.6910
10-Iso	150	24.9640	24.9722	24.8716

Table A5.1 Bending moment  $m_{21}$  vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

	MPa	kN.m	kN.m	kN.m
item	E soil	D group	P group	S group
1-Iso	5	-20.0225	-19.8812	-18.9020
2-Iso	10	-26.0066	-25.9861	-25.2084
3-Iso	15	-28.8234	-28.8328	-28.2145
4-Iso	20	-30.4588	-30.4774	-29.9682
5-Iso	30	-32.2790	-32.3009	-31.9264
6-Iso	40	-33.2683	-33.2888	-32.9933
7-Iso	50	-33.8898	-33.9083	-33.6645
8-Iso	70	-34.6275	-34.6426	-34.4621
9-Iso	120	-35.4288	-35.4389	-35.3295
10-Iso	150	-35.65952	-35.66788	-35.57937

Table A5.2 Bending moment  $m_{32}$  vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

	MPa	kN.m	kN.m	kN.m
item	E soil	D group	P group	S group
1-Iso	5	33.9010	33.2873	33.0020
2-Iso	10	33.1011	32.7724	32.6155
3-Iso	15	32.8389	32.6142	32.5049
4-Iso	20	32.7102	32.5396	32.4555
5-Iso	30	32.5839	32.4687	32.4110
6-Iso	40	32.5219	32.4349	32.3910
7-Iso	50	32.4851	32.4152	32.3798
8-Iso	70	32.4435	32.3934	32.3678
9-Iso	120	32.4007	32.3713	32.3562
10-Iso	150	32.3888	32.3653	32.3531

Table A5.3 Bending moment  $m_{34}$  vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

Data for Graphs 5.1, 5.2 & 5.3

**Nodal Displacements ( $\Delta_{x1}, \Delta_{y1}, \Phi_{z1}$ ) vs Isotropic soil Modulus of Elasticity ( $E_H$ )**  
( see Table 5.1a)  $\nu = 0.25$

	MPa		kN.m		kN.m		kN.m
item	E soil		D group		P group		S group
1-Iso	5		0.0025		0.0033		0.0068
2-Iso	10		0.0013		0.0017		0.0035
3-Iso	15		0.0009		0.0012		0.0024
4-Iso	20		0.0007		0.0009		0.0018
5-Iso	30		0.0005		0.0006		0.0012
6-Iso	40		0.0004		0.0005		0.0009
7-Iso	50		0.0003		0.0004		0.0007
8-Iso	70		0.0002		0.0003		0.0005
9-Iso	120		0.0001		0.0002		0.0003
10-Iso	150		0.0001		0.0001		0.0003

Table A5.4 Nodal displacement ( $\Delta_{x1}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

	MPa		kN.m		kN.m		kN.m
item	E soil		D group		P group		S group
1-Iso	5		-0.00404		-0.00515		-0.00872
2-Iso	10		-0.00163		-0.00221		-0.00397
3-Iso	15		-0.00097		-0.00136		-0.00253
4-Iso	20		-0.00067		-0.00097		-0.00184
5-Iso	30		-0.00041		-0.00061		-0.00119
6-Iso	40		-0.00029		-0.00045		-0.00088
7-Iso	50		-0.00023		-0.00035		-0.00069
8-Iso	70		-0.00016		-0.00024		-0.00049
9-Iso	120		-0.00009		-0.00014		-0.00028
10-Iso	150		-0.00007		-0.00011		-0.00022

Table A5.5 Nodal displacement ( $\Delta_{y1}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

	MPa		rad		rad		rad
item	E soil		D group		P group		S group
1-Iso	5		-0.00998		-0.00990		-0.00981
2-Iso	10		-0.00929		-0.00921		-0.00920
3-Iso	15		-0.00905		-0.00902		-0.00899
4-Iso	20		-0.00893		-0.00891		-0.00888
5-Iso	30		-0.00881		-0.00879		-0.00877
6-Iso	40		-0.00874		-0.00873		-0.00872
7-Iso	50		-0.00871		-0.00870		-0.00868
8-Iso	70		-0.00866		-0.00866		-0.00865
9-Iso	120		-0.00862		-0.00861		-0.00861
10-Iso	150		-0.00860		-0.00860		-0.00860

Table A5.6 Nodal Rotation ( $\Phi_{z1}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

Data for Graphs 5.4, 5.5 & 5.6



**Nodal displacements ( $\Delta x_3, \Delta y_3, \Phi_{z3}$ ) vs Isotropic soil modulus of elasticity( $E_H$ )**  
(see Table 5.1a)  $\nu = 0.25$

	MPa		kN.m		kN.m		kN.m
item	E soil		D group		P group		S group
1-Iso	5		0.02029		0.02085		0.02411
2-Iso	10		0.01702		0.01730		0.01896
3-Iso	15		0.01588		0.01606		0.01718
4-Iso	20		0.01529		0.01543		0.01627
5-Iso	30		0.01469		0.01478		0.01534
6-Iso	40		0.01438		0.01445		0.01488
7-Iso	50		0.01420		0.01425		0.01459
8-Iso	70		0.01399		0.01402		0.01427
9-Iso	120		0.01376		0.01378		0.01392
10-Iso	150		0.01370		0.01372		0.01383

Table A5.7 Nodal displacement ( $\Delta_{x3}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

	MPa		kN.m		kN.m		kN.m
item	E soil		D group		P group		S group
1-Iso	5		-0.02561		-0.02641		-0.03052
2-Iso	10		-0.01838		-0.01878		-0.02089
3-Iso	15		-0.01583		-0.01609		-0.01751
4-Iso	20		-0.01451		-0.01470		-0.01578
5-Iso	30		-0.01316		-0.01329		-0.01401
6-Iso	40		-0.01247		-0.01257		-0.01311
7-Iso	50		-0.01205		-0.01213		-0.01257
8-Iso	70		-0.01157		-0.01162		-0.01194
9-Iso	120		-0.01106		-0.01109		-0.01128
10-Iso	150		-0.01092		-0.01094		-0.01109

Table A5.8 Nodal displacement ( $\Delta_{y3}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

	MPa		rad		rad		rad
item	E soil		D group		P group		S group
1-Iso	5		-0.00469		-0.00460		-0.00470
2-Iso	10		-0.00384		-0.00379		-0.00384
3-Iso	15		-0.00353		-0.00350		-0.00353
4-Iso	20		-0.00337		-0.00335		-0.00337
5-Iso	30		-0.00321		-0.00319		-0.00321
6-Iso	40		-0.00312		-0.00311		-0.00313
7-Iso	50		-0.00307		-0.00306		-0.00307
8-Iso	70		-0.00301		-0.00301		-0.00301
9-Iso	120		-0.00295		-0.00295		-0.00295
10-Iso	150		-0.00293		-0.00293		-0.00293

Table A5.9 Nodal rotation ( $\Phi_{z3}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.25$

Data for Graphs 5.7, 5.8 & 5.9

**Bending Moments ( $M_{21}$ ,  $M_{32}$ ,  $M_{34}$ ) vs Isotropic Soil Modulus of Elasticity ( $E_H$ )**  
(see Table 5.1a)  $\nu=0.43$

item	E soil	D group	P group	S group
11-Iso	5	15.5239	15.9095	15.0556
12-Iso	10	19.3920	19.6476	19.0668
13-Iso	15	21.1144	21.3072	20.8701
14-Iso	20	22.0898	22.2446	21.8947
15-Iso	30	23.1557	23.2667	23.0170
16-Iso	40	23.7266	23.8132	23.6191
17-Iso	50	24.0825	24.1534	23.9947
18-Iso	70	24.5020	24.5541	24.4379
19-Iso	120	24.9543	24.9857	24.9160
20-Iso	150	25.0839	25.1092	25.0530

Table A5.10 Bending moment  $m_{21}$  vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.43$

item	E soil	D group	P group	S group
11-Iso	5	-20.6390	-20.4257	-19.5124
12-Iso	10	-26.6928	-26.6646	-26.0234
13-Iso	15	-29.4209	-29.4318	-28.9428
14-Iso	20	-30.9724	-30.9945	-30.6002
15-Iso	30	-32.6724	-32.6980	-32.4143
16-Iso	40	-33.5849	-33.6086	-33.3873
17-Iso	50	-34.1541	-34.1755	-33.9941
18-Iso	70	-34.8258	-34.8431	-34.7098
19-Iso	120	-35.5507	-35.5621	-35.4820
20-Iso	150	-35.7585	-35.7679	-35.7033

Table A5.11 Bending moment  $m_{32}$  vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.43$

item	E soil	D group	P group	S group
11-Iso	5	32.5682	31.7867	31.5535
12-Iso	10	32.4058	32.0054	31.8770
13-Iso	15	32.3694	32.1001	32.0108
14-Iso	20	32.3560	32.1532	32.0846
15-Iso	30	32.3466	32.2107	32.1638
16-Iso	40	32.3435	32.2413	32.2057
17-Iso	50	32.3422	32.2603	32.2316
18-Iso	70	32.3412	32.2827	32.2619
19-Iso	120	32.3409	32.3067	32.2945
20-Iso	150	32.3410	32.3136	32.3038

Table A5.12 Bending moment  $m_{34}$  vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.43$

Data for Graphs 5.10, 5.11 & 5.12

· **Nodal Displacements ( $\Delta x_1, \Delta y_1, \Phi_{z1}$ ) vs Modulus of Elasticity ( $E_H$ ) Isotropic soil**  
(see Table 5.1a)  $\nu = 0.43$

item	E soil	D group	P group	S group
11-Iso	5	0.0008	0.0014	0.0047
12-Iso	10	0.0004	0.0007	0.0024
13-Iso	15	0.0003	0.0005	0.0016
14-Iso	20	0.0002	0.0004	0.0012
15-Iso	30	0.0002	0.0002	0.0008
16-Iso	40	0.0001	0.0002	0.0006
17-Iso	50	0.0001	0.0002	0.0005
18-Iso	70	0.0001	0.0001	0.0004
19-Iso	120	0.0000	0.0001	0.0002
20-Iso	150	0.0000	0.0001	0.0002

Table A5.13 Nodal displacement ( $\Delta_{x1}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.43$

item	E soil	D group	P group	S group
11-Iso	5	-0.0034	-0.0044	-0.0076
12-Iso	10	-0.0014	-0.0019	-0.0035
13-Iso	15	-0.0008	-0.0012	-0.0022
14-Iso	20	-0.0006	-0.0008	-0.0016
15-Iso	30	-0.0004	-0.0005	-0.0011
16-Iso	40	-0.0003	-0.0004	-0.0008
17-Iso	50	-0.0002	-0.0003	-0.0006
18-Iso	70	-0.0001	-0.0002	-0.0004
19-Iso	120	-0.0001	-0.0001	-0.0003
20-Iso	150	-0.0001	-0.0001	-0.0002

Table A5.14 Nodal displacement ( $\Delta_{y1}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.43$

item	E soil	D group	P group	S group
11-Iso	5	-0.00993	-0.00992	-0.00993
12-Iso	10	-0.00927	-0.00927	-0.00927
13-Iso	15	-0.00904	-0.00904	-0.00904
14-Iso	20	-0.00893	-0.00892	-0.00892
15-Iso	30	-0.00880	-0.00880	-0.00880
16-Iso	40	-0.00874	-0.00874	-0.00874
17-Iso	50	-0.00870	-0.00870	-0.00870
18-Iso	70	-0.00866	-0.00866	-0.00866
19-Iso	120	-0.00862	-0.00862	-0.00861
20-Iso	150	-0.00860	-0.00860	-0.00860

Table A5.15 Nodal rotation ( $\Phi_{z1}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.43$

Data for Graphs 5.13, 5.14 & 5.15

**Nodal Displacements ( $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$ ) vs Modulus of Elasticity ( $E_H$ ) Isotropic soil**  
(see Table 5.1a)  $\nu = 0.43$

item	E soil	D group	P group	S group
11-Iso	5	0.01846	0.01902	0.02232
12-Iso	10	0.01608	0.01636	0.01803
13-Iso	15	0.01524	0.01543	0.01654
14-Iso	20	0.01481	0.01495	0.01578
15-Iso	30	0.01437	0.01446	0.01502
16-Iso	40	0.01414	0.01421	0.01463
17-Iso	50	0.01400	0.01406	0.01439
18-Iso	70	0.01384	0.01388	0.01412
19-Iso	120	0.01368	0.01370	0.01384
20-Iso	150	0.01363	0.01365	0.01376

Table A5.16 Nodal displacement ( $\Delta_{x3}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.43$

item	E soil	D group	P group	S group
11-Iso	5	-0.02378	-0.02450	-0.02819
12-Iso	10	-0.01740	-0.01775	-0.01963
13-Iso	15	-0.01515	-0.01538	-0.01664
14-Iso	20	-0.01399	-0.01416	-0.01511
15-Iso	30	-0.01280	-0.01292	-0.01356
16-Iso	40	-0.01220	-0.01228	-0.01277
17-Iso	50	-0.01184	-0.01190	-0.01229
18-Iso	70	-0.01141	-0.01146	-0.01174
19-Iso	120	-0.01097	-0.01100	-0.01116
20-Iso	150	-0.01084	-0.01087	-0.01100

Table A5.17 Nodal displacement ( $\Delta_{y3}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.43$

item	E soil	D group	P group	S group
11-Iso	5	-0.00441	-0.00431	-0.00438
12-Iso	10	-0.00368	-0.00363	-0.00367
13-Iso	15	-0.00342	-0.00339	-0.00341
14-Iso	20	-0.00329	-0.00326	-0.00328
15-Iso	30	-0.00315	-0.00313	-0.00315
16-Iso	40	-0.00308	-0.00307	-0.00308
17-Iso	50	-0.00304	-0.00303	-0.00304
18-Iso	70	-0.00299	-0.00298	-0.00299
19-Iso	120	-0.00294	-0.00293	-0.00294
20-Iso	150	-0.00292	-0.00292	-0.00292

Table A5.18 Nodal rotation ( $\Phi_{z3}$ ) vs Isotropic soil modulus of elasticity  $E_H$  for  $\nu=0.43$

Data for Graphs 5.16, 5.17 & 5.18

# A5.2

Tables associated with  
Graphs 5.19-5.36  
Output of structural analysis  
associated with  
cross-anisotropic soil  
(sub-cases 3 & 5 from Table 5.1)

**Bending Moment ( $M_{21}$ ,  $M_{32}$ ,  $M_{34}$ ) vs Cross-anisotropic soil Modulus of Elasticity ( $E_v$ )**  
(See Table 5.1b)

item	$E_v$ (MPa)	D group	P group	S group
1-Ciso	26	21.8510	21.8785	21.4490
2-Ciso	27	21.9743	22.0015	21.5854
3-Ciso	28	22.0898	22.1167	21.7131
4-Ciso	29	22.1982	22.2247	21.8329
5-Ciso	30	22.3001	22.3263	21.9456
6-Ciso	31	22.3961	22.4219	22.0518
7-Ciso	32	22.4867	22.5121	22.1520
8-Ciso	33	22.5723	22.5974	22.2467
9-Ciso	34	22.6534	22.6781	22.3364
10-Ciso	35	22.7303	22.7546	22.4214

Table A5.19 Bending moment  $M_{21}$  vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
1-Ciso	26	-32.4010	-32.4440	-32.0647
2-Ciso	27	-32.5373	-32.5796	-32.2114
3-Ciso	28	-32.6651	-32.7065	-32.3489
4-Ciso	29	-32.7850	-32.8258	-32.4781
5-Ciso	30	-32.8979	-32.9380	-32.5997
6-Ciso	31	-33.0043	-33.0437	-32.7144
7-Ciso	32	-33.1048	-33.1435	-32.8227
8-Ciso	33	-33.1999	-33.2378	-32.9251
9-Ciso	34	-33.2899	-33.3272	-33.0221
10-Ciso	35	-33.3753	-33.4120	-33.1142

Table A5.20 Bending moment  $M_{32}$  vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
1-Ciso	26	33.3255	33.2285	33.1741
2-Ciso	27	33.2903	33.1966	33.1441
3-Ciso	28	33.2575	33.1670	33.1162
4-Ciso	29	33.2269	33.1393	33.0902
5-Ciso	30	33.1983	33.1134	33.0659
6-Ciso	31	33.1714	33.0892	33.0431
7-Ciso	32	33.1462	33.0664	33.0217
8-Ciso	33	33.1225	33.0449	33.0016
9-Ciso	34	33.1001	33.0247	32.9826
10-Ciso	35	33.0790	33.0057	32.9647

Table A5.21 Bending moment  $M_{34}$  vs Cross-anisotropic soil modulus of elasticity  $E_v$

Data for Graphs 5.19, 5.20 & 5.21

**Nodal Displacements ( $\Delta x_1, \Delta y_1, \Phi_{z1}$ ) vs Cross-anisotropic modulus of Elasticity ( $E_v$ )**  
(See Table 5.1b)

item	$E_v$ (MPa)	D group	P group	S group
1-Ciso	26	0.00140	0.00153	0.00208
2-Ciso	27	0.00135	0.00147	0.00201
3-Ciso	28	0.00130	0.00142	0.00194
4-Ciso	29	0.00126	0.00137	0.00188
5-Ciso	30	0.00122	0.00133	0.00182
6-Ciso	31	0.00118	0.00129	0.00176
7-Ciso	32	0.00115	0.00125	0.00171
8-Ciso	33	0.00111	0.00121	0.00165
9-Ciso	34	0.00108	0.00118	0.00161
10-Ciso	35	0.00105	0.00115	0.00156

Table A5.22 Nodal displacement ( $\Delta x_1$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
1-Ciso	26	-0.00045	-0.00067	-0.00130
2-Ciso	27	-0.00043	-0.00064	-0.00124
3-Ciso	28	-0.00041	-0.00062	-0.00120
4-Ciso	29	-0.00040	-0.00059	-0.00115
5-Ciso	30	-0.00038	-0.00057	-0.00111
6-Ciso	31	-0.00037	-0.00055	-0.00107
7-Ciso	32	-0.00035	-0.00053	-0.00104
8-Ciso	33	-0.00034	-0.00051	-0.00101
9-Ciso	34	-0.00033	-0.00050	-0.00097
10-Ciso	35	-0.00032	-0.00048	-0.00094

Table A5.23 Nodal displacement ( $\Delta y_1$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
1-Ciso	26	-0.00872	-0.00871	-0.00872
2-Ciso	27	-0.00872	-0.00870	-0.00869
3-Ciso	28	-0.00871	-0.00870	-0.00868
4-Ciso	29	-0.00871	-0.00869	-0.00868
5-Ciso	30	-0.00870	-0.00869	-0.00867
6-Ciso	31	-0.00870	-0.00868	-0.00867
7-Ciso	32	-0.00869	-0.00868	-0.00866
8-Ciso	33	-0.00869	-0.00867	-0.00866
9-Ciso	34	-0.00868	-0.00867	-0.00866
10-Ciso	35	-0.00868	-0.00867	-0.00865

Table A5.24 Nodal rotation ( $\Phi_{z1}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

Data for Graphs 5.22, 5.23 & 5.24

**Nodal Displacements ( $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$ ) vs Cross-anisotropic Modulus of Elasticity ( $E_v$ )**  
(see Table 5.1b)

item	$E_v$ (MPa)	D group	P group	S group
1-Ciso	26	0.01539	0.01547	0.01595
2-Ciso	27	0.01532	0.01539	0.01589
3-Ciso	28	0.01525	0.01533	0.01580
4-Ciso	29	0.01519	0.01526	0.01572
5-Ciso	30	0.01513	0.01520	0.01565
6-Ciso	31	0.01508	0.01515	0.01558
7-Ciso	32	0.01503	0.01509	0.01551
8-Ciso	33	0.01498	0.01504	0.01545
9-Ciso	34	0.01494	0.01500	0.01539
10-Ciso	35	0.01490	0.01495	0.01534

Table A5.25 Nodal displacement ( $\Delta_{x3}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
1-Ciso	26	-0.01335	-0.01349	-0.01426
2-Ciso	27	-0.01324	-0.01338	-0.01413
3-Ciso	28	-0.01314	-0.01327	-0.01400
4-Ciso	29	-0.01305	-0.01317	-0.01388
5-Ciso	30	-0.01296	-0.01308	-0.01376
6-Ciso	31	-0.01288	-0.01300	-0.01366
7-Ciso	32	-0.01280	-0.01291	-0.01355
8-Ciso	33	-0.01273	-0.01284	-0.01346
9-Ciso	34	-0.01266	-0.01277	-0.01337
10-Ciso	35	-0.01260	-0.01270	-0.01328

Table A5.26 Nodal displacement ( $\Delta_{y3}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
1-Ciso	26	-0.00327	-0.00326	-0.00327
2-Ciso	27	-0.00326	-0.00324	-0.00326
3-Ciso	28	-0.00325	-0.00323	-0.00325
4-Ciso	29	-0.00323	-0.00322	-0.00324
5-Ciso	30	-0.00322	-0.00321	-0.00322
6-Ciso	31	-0.00321	-0.00320	-0.00321
7-Ciso	32	-0.00320	-0.00319	-0.00320
8-Ciso	33	-0.00319	-0.00318	-0.00319
9-Ciso	34	-0.00318	-0.00317	-0.00318
10-Ciso	35	-0.00317	-0.00316	-0.00317

Table A5.27 Nodal rotation ( $\Phi_{z3}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

Data for Graphs 5.25, 5.26 & 5.27



**Bending Moment ( $M_{21}$ ,  $M_{32}$ ,  $M_{34}$ ) vs Cross-anisotropic soil Modulus of Elasticity ( $E_v$ )**  
(See Table 5.1b)

item	$E_v$ (MPa)	D group	P group	S group
21-Ciso	75	24.3209	24.3358	24.1869
22-Ciso	76	24.3373	24.3520	24.2049
23-Ciso	77	24.3532	24.3678	24.2224
4-Ciso	78	24.3687	24.3831	24.2396
25-Ciso	79	24.3839	24.3982	24.2563
26-Ciso	80	24.3987	24.4128	24.2726
27-Ciso	81	24.4131	24.4271	24.2885
28-Ciso	82	24.4272	24.4410	24.3041
29-Ciso	83	24.4410	24.4547	24.3193
30-Ciso	84	24.4544	24.4680	24.3341

Table A5.28 Bending moment  $M_{21}$  vs Cross-anisotropic soil modulus of elasticity  $E_v$

**Bending Moment  $M_{32}$  vs Modulus of Elasticity ( $E_v$ ) Cross-anisotropic medium**

item	$E_v$ (MPa)	D group	P group	S group
21-Ciso	75	-35.0783	-35.1026	-34.9631
22-Ciso	76	-35.0976	-35.1215	-34.9838
23-Ciso	77	-35.1164	-35.1401	-35.0040
4-Ciso	78	-35.1347	-35.1581	-35.0237
25-Ciso	79	-35.1526	-35.1757	-35.0429
26-Ciso	80	-35.1700	-35.1929	-35.0616
27-Ciso	81	-35.1871	-35.2097	-35.0799
28-Ciso	82	-35.2037	-35.2261	-35.0978
29-Ciso	83	-35.2200	-35.2422	-35.1153
30-Ciso	84	-35.2359	-35.2578	-35.1323

Table A5.29 Bending moment  $M_{32}$  vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
21-Ciso	75	32.6218	32.5930	32.5762
22-Ciso	76	32.6181	32.5897	32.5732
23-Ciso	77	32.6146	32.5865	32.5702
24-Ciso	78	32.6111	32.5834	32.5673
25-Ciso	79	32.6077	32.5803	32.5644
26-Ciso	80	32.6044	32.5774	32.5617
27-Ciso	81	32.6012	32.5745	32.5590
28-Ciso	82	32.5981	32.5717	32.5563
29-Ciso	83	32.5950	32.5689	32.5538
30-Ciso	84	32.5920	32.5662	32.5513

Table A5.30 Bending moment  $M_{34}$  vs Cross-anisotropic soil modulus of elasticity  $E_v$

Data for Graphs 5.28, 5.29 & 5.30

**Nodal Displacement ( $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$ ) vs Modulus of Elasticity ( $E_v$ ) of Cross-anisotropic medium (See Table 5.1b)**

item	$E_v$ (MPa)	D group	P group	S group
21-Ciso	75	0.00041	0.00045	0.00061
22-Ciso	76	0.00040	0.00044	0.00060
23-Ciso	77	0.00040	0.00044	0.00060
4-Ciso	78	0.00039	0.00043	0.00059
25-Ciso	79	0.00039	0.00042	0.00058
26-Ciso	80	0.00038	0.00042	0.00057
27-Ciso	81	0.00038	0.00041	0.00057
28-Ciso	82	0.00037	0.00041	0.00056
29-Ciso	83	0.00037	0.00040	0.00055
30-Ciso	84	0.00037	0.00040	0.00055

Table A5.31 Nodal displacement ( $\Delta_{x1}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
21-Ciso	75	-0.00012	-0.00020	-0.00040
22-Ciso	76	-0.00012	-0.00020	-0.00040
23-Ciso	77	-0.00012	-0.00019	-0.00039
24-Ciso	78	-0.00012	-0.00019	-0.00039
25-Ciso	79	-0.00012	-0.00019	-0.00038
26-Ciso	80	-0.00012	-0.00019	-0.00038
27-Ciso	81	-0.00011	-0.00018	-0.00037
28-Ciso	82	-0.00011	-0.00018	-0.00037
29-Ciso	83	-0.00011	-0.00018	-0.00036
30-Ciso	84	-0.00011	-0.00018	-0.00036

Table A5.32 Nodal displacement ( $\Delta_{y1}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
21-Ciso	75	-0.00861	-0.00861	-0.00860
22-Ciso	76	-0.00860	-0.00861	-0.00861
23-Ciso	77	-0.00861	-0.00861	-0.00860
24-Ciso	78	-0.00861	-0.00861	-0.00860
25-Ciso	79	-0.00861	-0.00860	-0.00860
26-Ciso	80	-0.00861	-0.00860	-0.00860
27-Ciso	81	-0.00861	-0.00860	-0.00860
28-Ciso	82	-0.00861	-0.00860	-0.00860
29-Ciso	83	-0.00861	-0.00860	-0.00860
30-Ciso	84	-0.00861	-0.00860	-0.00860

Table A5.33 Nodal rotation ( $\Phi_{z1}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

Data for Graphs 5.31, 5.32 & 5.33

**Nodal Displacements ( $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$ ) vs Cross-anisotropic Modulus of Elasticity ( $E_v$ )**  
(see Table 5.1b)

item	$E_v$ (MPa)	D group	P group	S group
21-Ciso	75	0.01405	0.01407	0.01422
22-Ciso	76	0.01404	0.01406	0.01421
23-Ciso	77	0.01403	0.01405	0.01420
24-Ciso	78	0.01402	0.01404	0.01419
25-Ciso	79	0.01402	0.01404	0.01418
26-Ciso	80	0.01401	0.01403	0.01417
27-Ciso	81	0.01400	0.01402	0.01416
28-Ciso	82	0.01400	0.01401	0.01415
29-Ciso	83	0.01399	0.01401	0.01414
30-Ciso	84	0.01398	0.01400	0.01414

Table A5.34 Nodal displacement ( $\Delta_{x3}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
21-Ciso	75	-0.01135	-0.01139	-0.01165
22-Ciso	76	-0.01133	-0.01138	-0.01163
23-Ciso	77	-0.01132	-0.01136	-0.01162
24-Ciso	78	-0.01131	-0.01135	-0.01160
25-Ciso	79	-0.01130	-0.01134	-0.01159
26-Ciso	80	-0.01128	-0.01133	-0.01157
27-Ciso	81	-0.01127	-0.01131	-0.01155
28-Ciso	82	-0.01126	-0.01130	-0.01154
29-Ciso	83	-0.01125	-0.01129	-0.01153
30-Ciso	84	-0.01124	-0.01128	-0.01151

Table A5.35 Nodal displacement ( $\Delta_{y3}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

item	$E_v$ (MPa)	D group	P group	S group
21-Ciso	75	-0.00300	-0.00299	-0.00300
23-Ciso	77	-0.00299	-0.00299	-0.00300
4-Ciso	78	-0.00299	-0.00299	-0.00299
25-Ciso	79	-0.00299	-0.00299	-0.00299
26-Ciso	80	-0.00299	-0.00298	-0.00299
27-Ciso	81	-0.00299	-0.00298	-0.00299
28-Ciso	82	-0.00299	-0.00298	-0.00299
29-Ciso	83	-0.00299	-0.00298	-0.00299
30-Ciso	84	-0.00298	-0.00298	-0.00299

Table A5.36 Nodal rotation ( $\Phi_{z3}$ ) vs Cross-anisotropic soil modulus of elasticity  $E_v$

Data for Graphs 5.34, 5.35 & 5.36

# A5.3

Tables associated with  
Graphs 5.37-5.45  
Output of structural analysis  
associated with  
cross-anisotropic soil  
(sub-cases 7 to 11 from Table 5.1)

**Bending Moment ( $M_{21}$ ,  $M_{32}$ ,  $M_{34}$ ) vs Cross-anisotropic soil Modulus of Elasticity ( $F_v$ )**  
(See Table 5.1c)

item	$F_v$ (MPa)	D group	P group	S group
35-Ciso*	21.429	22.129	22.218	21.941
32-Ciso	21.450	22.370	22.466	22.232
34-Ciso	22.627	22.439	22.532	22.301
33-Ciso	23.000	22.460	22.561	22.342

Table A5.37 Bending moment  $M_{21}$  vs Corss-anisotropic soil modulus of elasticity  $F_v$

item	$F_v$ (MPa)	D group	P group	S group
35-Ciso*	21.429	-33.229	-33.258	-32.958
32-Ciso	21.450	-33.629	-33.652	-33.384
34-Ciso	22.627	-33.689	-33.712	-33.449
33-Ciso	23.000	-33.707	-33.733	-33.476

Table A5.38 Bending moment  $M_{32}$  vs Corss-anisotropic soil modulus of elasticity  $F_v$

item	$F_v$ (MPa)	D group	P group	S group
35-Ciso*	21.429	33.518	33.394	33.344
32-Ciso	21.450	33.519	33.395	33.348
34-Ciso	22.627	33.491	33.370	33.324
33-Ciso	23.000	33.482	33.361	33.315

Table A5.39 Bending moment  $M_{34}$  vs Corss-anisotropic soil modulus of elasticity  $F_v$

\*: Cross-anisotropy simplifies to isotropy.

Data for Graphs 5.37, 5.38 & 5.39

**Nodal Displacement ( $\Delta_{x1}$ ,  $\Delta_{y1}$ ,  $\Phi_{z1}$ ) vs Modulus of Elasticity ( $F_v$ ) of Cross-anisotropic  $\lambda$**   
(See Table 5.1c)

item	$F_v$ (MPa)	D group	P group	S group
35-Ciso*	21.429	0.0016	0.0017	0.0023
32-Ciso	21.450	0.0016	0.0017	0.0023
34-Ciso	22.627	0.0016	0.0017	0.0022
33-Ciso	23.000	0.0016	0.0017	0.0022

Table A5.40 Nodal displacement ( $\Delta_{x1}$ ) vs Cross-anisotropic soil modulus of elasticity  $F_v$

item	$F_v$ (MPa)	D group	P group	S group
35-Ciso*	21.429	-0.0004	-0.0006	-0.0011
32-Ciso	21.450	-0.0003	-0.0005	-0.0010
34-Ciso	22.627	-0.0003	-0.0005	-0.0009
33-Ciso	23.000	-0.0003	-0.0005	-0.0009

Table A5.41 Nodal displacement ( $\Delta_{y1}$ ) vs Cross-anisotropic soil modulus of elasticity  $F_v$

item	$F_v$ (MPa)	D group	P group	S group
35-Ciso*	21.429	-0.00864	-0.00861	-0.00863
32-Ciso	21.450	-0.00861	-0.00861	-0.00861
34-Ciso	22.627	-0.00861	-0.00861	-0.00861
33-Ciso	23.000	-0.00861	-0.00861	-0.00861

Table A5.42 Nodal rotation ( $\Phi_{z1}$ ) vs Cross-anisotropic soil modulus of elasticity  $F_v$

\*: Cross-anisotropy  $\lambda$ simplifies to isotropy.

Data for Graphs 5.40, 5.41 & 5.42

**Nodal Displacements ( $\Delta_{x3}$ ,  $\Delta_{y3}$ ,  $\Phi_{z3}$ ) vs Cross-anisotropic Modulus of Elasticity ( $F_v$ )**  
(see Table 5.1c)

item	$F_v$ (MPa)	D group	P group	S group
35-Ciso*	21.429	0.01537	0.01546	0.01603
32-Ciso	21.450	0.01527	0.01536	0.01591
34-Ciso	22.627	0.01530	0.01532	0.01585
33-Ciso	23.000	0.01522	0.01530	0.01583

Table A5.43 Nodal displacement ( $\Delta_{x3}$ ) vs Cross-anisotropic soil modulus of elasticity  $F_v$

item	$F_v$ (MPa)	D group	P group	S group
35-Ciso*	21.429	-0.01285	-0.01297	-0.01363
32-Ciso	21.450	-0.01258	-0.01269	-0.01327
34-Ciso	22.627	-0.01253	-0.01263	-0.01321
33-Ciso	23.000	-0.01252	-0.01262	-0.01319

Table A5.44 Nodal displacement ( $\Delta_{y3}$ ) vs Cross-anisotropic soil modulus of elasticity  $F_v$

item	$F_v$ (MPa)	D group	P group	S group
35-Ciso*	21.429	-0.00323	-0.00321	-0.00322
32-Ciso	21.450	-0.00320	-0.00318	-0.00319
34-Ciso	22.627	-0.00319	-0.00317	-0.00318
33-Ciso	23.000	-0.00319	-0.00317	-0.00318

Table A5.45 Nodal rotation ( $\Phi_{z3}$ ) vs Cross-anisotropic soil modulus of elasticity  $F_v$

\*: Cross-anisotropy \simplifies to isotropy.

Data for Graphs 5.43, 5.44 & 5.45